m0n0wall 中文手册

作者:

Chris Buechler
Manuel Kasper

译文：Ben Zeng

m0n0wall 的作者是 Manuel Kasper。大部分文档由 Chris Buechler 编写。其他贡献者在贡献者列表中。

m0nowall 版本 1.2

中的， 即使已被告知发生该损失的可能性。

2005 年 9 月

摘要

一个可自由再分发的、完整的、嵌入式的防火墙软件包。

目录

第一章．介绍

1.1 m0n0wall 是什么

m0n0wall 是一项针对建立一个完整的、嵌入式的防火墙软件包的计划，该软件包可以安装于嵌入式 PC 里， 提供所有商业防火墙的重要特性（包括易用性），而且价格只有商业防火墙几分之一（自由软件）。

m0n0wall 是基于 bare-bones version of FreeBSD，包括一个 WEB 服务器，PHP 和其它一些工具软件。整个系统的配置保存在一个 XML 文件当中，条理清晰。

m0n0wall 可能是第一个启动时通过 PHP 配置的 UNIX 系统，这种结构胜于使用 shell 脚本。并且整个系统的配置用 XML 格式保存。

1.2 m0n0wall 不是什么

m0n0 是一个防火墙，而防火墙的目的是提供安全。增加越多的功能，新增功能的弱点给防火墙带来安全隐患的机会就越大。m0n0wall 创建者及主要贡献者的观点是防火墙第 3 和 4 层基本服务之外的任何东西都不属于 m0n0wall。一些可能合适的服务占用 CPU 和内存饥渴，而 m0n0wall 着眼于嵌入式设备，CPU 和内存资源都有限。非连续（保存）的文件系统是由于着眼于 CF(Compact Flash)安装,这又是一个限制因素。最后，(内核)映象的大小限制，消除了其它可能性。

我们觉得以下服务应该运行在其它服务器，并特意不作为 m0n0wall 的一部分：

- 入侵检测/保护系统(IDS)
- 代理服务器
- 第三、第四层外的任何数据包检查
- 通用的 WEB 服务器
- FTP 服务器
- 网络时间服务器
- 日志文件分析器

基于同样的原因 ,m0n0 不允许登录(login): 控制台没有登录提示符(以显示一个菜单代替）,没有 telnet 和 ssh 服务进程(deamon)

1.3 历史

Manuel Kasper, m0n0wall 的作者，说：

从我开始在嵌入式 PC 上摆弄包过滤器，我就想有一个漂亮的基于 web 图形界面的

控制器来控制所有的防火墙功能，而不是通过键入单个的命令。在互联网上有很多漂亮的带有 WEB 接口的防火墙包（大部分是基于 Linux 的），但是没有一个符合我要求的（自由，快速，简单，干净以及我需要的所有特性）。所以，我终于开始写属于自己的 WEB 图形界面。但是，我决不是想建立一个 webmin 的翻版----我想建立一个完整的、新的嵌入式防火墙软件包。它的所有将被发展为一个接上电源的盒子，可以通过串口设置 LAN IP 地址，登录进 WEB 界面设置它。然后我决定我不能像平常的启动系统那样通过 SHELL 脚本配置系统（由于它几乎不可能用 SHELL 脚本完成，所以我已经写了一个 C 程序产生过滤器规则），并且自从我使用了基于 PHP 的 WEB 接口，不长时间我就发现还是使用 PHP 来配置系统的好。这种方法，配置数据将不再必须被存储在那些被 SHELL 脚本解析的文本文件里面----它现在被存储在一个 XML 文件里。所以我又完全重写了整个系统，除了相当多的“引擎罩底下的东西“外，看上去感觉没有什么改变。

m0n0wall 的第一个公共 beta 版于 2003 年 2 月 15 日，1.0 版本正好在一年后的 2004 年 2 月 15 日。这两个版本之间共发布了另外 26 个公共 beta 版，平均每两个星期发布一个。每个版本完整的修改列表可以在 m0n0wall 网站的 Change Log 栏目找到。

1.4 功能

monowall 提供了很多昂贵商用防火墙中的功能，其中一些功能还是商用防火墙中没有的。包括：

- WEB 界面（支持 SSL）
- 用于恢复系统的串口界面
    - 设置 LAN IP 地址

- 重置密码
- 恢复初始默认设置
- 重启系统

- 无线支持 （access point with PRISM-II/2.5/3 cards, BSS/IBSS with other cards including Cisco）
- 上网认证（captive portal）
- 支持 802.1Q VLAN
- 基于状态的包过滤
    - block/pass 规则
    - 日志
- NAT/PAT (包括 1:1)
- 在 WAN 口上支持 DHCP 客户、PPPoE、PPTP 和 Telstra BigPond Cable
- IPsec VPN 隧道（IKE; 支持硬件加密卡，移动客户和证书）
- PPTP VPN （支持 RADIUS 服务器）
- 静态路由
- DHCP 服务器与中继
- 缓存 DNS 转向器
- 动态 DNS 客户端与 RFC 2136 DNS 更新器
- SNMP 代理
- 流量整形（带宽限制）
- 基于 SVG 的流量图
- 可以通过 WEB 界面进行固件升级

- 唤醒 LAN 客户
- 配置文件备份/恢复
- 主机/网络别名

1.4.1 组件

m0n0wall 包括以下软件组件：

- FreeBSD components (kernel, user programs)
- ipfilter
- PHP (CGI version)
- thttpd
- MPD
- ISC DHCP server
- ez-ipupdate (for DynDNS updates)
- Dnsmasq (for the caching DNS forwarder)
- Raccoon(for IPsec IKE)

1.4.2 规格

- 当前的m0n0wall系统可以存放在小于 6M 的CF卡（或者CD-ROM）上
- 在 net4501 平台上，当运行默认配置的时候，包含 NAT 在内， m0n0wall 提供大约 17 Mbps 的 WAN <-> LAN TCP 吞吐量。 在更快的平台上（类似于 net4801 或者 WRAP），吞吐量可能超过 50Mbp（在更新的标准 PC 上 > 100 Mbps）。
- 在 net4501 平台上, m0n0wall 从上电启动到完全可以工作的时间小于 40 秒 ,这其中包含 POST (适当的 BIOS 配置）

1.5 软件版本和分发(许可)

允许以源代码和二进制形式再分发未经修改或经过修改的版本，只要满足以下的条件：

- 以源代码形式再分发必须不加修改地保留上述版权告示、本条件清单和下述弃权书。
- 以二进制形式再分发必须复制上述版权告示、本条件清单和下述弃权书到再分发相关的文档和其介质中。

本软件“按现状条件” 提供，并在此明示不提供任何明示或暗示的保障， 包括但不限于对商业适销性、对特定目的的适用性的暗示保障。 任何情况下，均不对任何直接、 间接、 偶然、 特殊、 惩罚性的， 或必然的损失 (包括但不限于替代商品或服务的采购、 使用、 数据或利益的损失或营业中断) 负责，无论是如何导致的并以任何有责任逻辑的， 无论是否是在本文档使用以外以任何方式产生的契约、严格责任或是民事侵权行为(包括疏忽或其它)中的， 即使已被告知发生该损失的可能性。

1.5.1 其它软件包

m0n0wall 基于或包含多种自由软件，在下面列出。m0n0wall 的作者很感谢这些软件作才所做出的努力。

FreeBSD (http://www.freebsd.org) Copyright © 1994-2003 FreeBSD, Inc. All rights reserved.

This product includes PHP, freely available from http://www.php.net. Copyright © 1999 - 2003 The PHP Group. All rights reserved.

mini_httpd (http://www.acme.com/software/mini_httpd) Copyright © 1999, 2000 by Jef Poskanzer <jef@acme.com>. All rights reserved.

ISC DHCP server (http://www.isc.org/products/DHCP) Copyright © 1996-2003 Internet Software Consortium. All rights reserved.

ipfilter (http://www.ipfilter.org) Copyright © 1993-2002 by Darren Reed.

MPD - Multi-link PPP daemon for FreeBSD (http://www.dellroad.org/mpd) Copyright © 1995-1999 Whistle Communications, Inc. All rights reserved.

ez-ipupdate (http://www.gusnet.cx/proj/ez-ipupdate) Copyright © 1998-2001 Angus Mackay. All rights reserved.

Circular log support for FreeBSD syslogd (http://software.wwwi.com/syslogd) Copyright © 2001 Jeff Wheelhouse (jdw@wwwi.com)

Dnsmasq - a DNS forwarder for NAT firewalls (http://www.thekelleys.org.uk) Copyright © 2000-2003 Simon Kelley

Racoon (http://www.kame.net/racoon) Copyright © 1995-2002 WIDE Project. All rights reserved.

before version pb23: watchdogd (watchdog) Copyright © 2002-2003 Dirk-Willem van Gulik. All rights reserved. This product includes software developed by the Stichting Wireless Leiden (http://www.wirelessleiden.nl). See LICENSE for more licensing information.

msntp (http://www.hpcf.cam.ac.uk/export) Copyright © 1996, 1997, 2000 N.M. Maclaren, University of Cambridge. All rights reserved.

UCD-SNMP (http://www.ece.ucdavis.edu/ucd-snmp) Copyright © 1989, 1991, 1992 by Carnegie Mellon University. Copyright © 1996, 1998-2000 The Regents of the University of California. All rights reserved. Copyright © 2001-2002, Network Associates Technology, Inc. All rights reserved. Portions of this code are copyright © 2001-2002, Cambridge Broadband Ltd. All rights reserved.

choparp (http://choparp.sourceforge.net) Copyright © 1997 Takamichi Tateoka (tree@mma.club.uec.ac.jp) Copyright © 2002 Thomas Quinot (thomas@cuivre.fr.eu.org)

1.6 贡献者列表

1.6.1 代码

m0n0wall 由 Manuel Kasper 编写。

下面人员为 m0n0wall 贡献了代码:

Bob Zoller (bob at kludgebox dot com): Diagnostics: Ping function; WLAN channel auto-select; DNS forwarder

Michael Mee (m0n0wall at mikemee dot com): Timezone and NTP client support

Magne Andreassen (magne dot andreassen at bluezone dot no): Remote syslog'ing; some code bits for DHCP server on optional interfaces

Rob Whyte (rob at g-labs dot com): Idea/code bits for encrypted webGUI passwords; minimalized SNMP agent

Petr Verner (verner at ipps dot cz): Advanced outbound NAT: destination selection

Bruce A. Mah (bmah at acm dot org): Filtering bridge patches

Jim McBeath (monowall at j dot jimmc dot org): Filter rule patches (ordering, block/pass, disabled); better status page; webGUI assign network ports page

Chris Olive (chris at technologEase dot com): enhanced "execute command" page

Pauline Middelink (middelink at polyware dot nl): DHCP client: send hostname patch

Björn Pålsson (bjorn at networksab dot com): DHCP lease list page

Peter Allgeyer (allgeyer at web dot de): "reject" type filter rules

Thierry Lechat (dev at lechat dot org): SVG-based traffic grapher

Steven Honson (steven at honson dot org): per-user IP address assignments for PPTP VPN

Kurt Inge Smådal (kurt at emsp dot no): NAT on optional interfaces

Dinesh Nair (dinesh at alphaque dot com): captive portal: pass-through MAC/IP addresses, RADIUS authentication HTTP server concurrency limit

Justin Ellison (justin at techadvise dot com): traffic shaper TOS matching; magic shaper; DHCP deny unknown clients; IPsec user FQDNs

Fred Wright (fw at well dot com): ipfilter window scaling fix; ipnat ICMP checksum adjustment fix

1.6.2 文档

m0n0wall 由 Manuel Kasper 编写。

下面人员为 m0n0wall 贡献了文档:

Chris Buechler (m0n0wall at chrisbuechler.com): Editor, numerous contributions throughout.

Jim McBeath (monowall at j dot jimmc dot org): Users Guide outline, editing

Rudi van Drunen (r.van.drunen at xs4all dot nl) with thanks to Manuel Kasper, Edwin Kremer, PicoBSD, Matt Simerson and John Voight: m0n0wall Hackers Guide, used as the basis for the Development chapter.

Francisco Artes (falcor at netassassin.com): IPsec and PPTP chapters.

Fred Wright (fw at well dot com): Suggestions and review.

Axel Eble (axel+m0n0-0001 at balrog dot de): Help with the wiki, ddclient howto contribution.

Brian Zushi (brian at ricerage dot org): Linux CD burning instructions, documentation review and suggestions.

Dino Bijedic (dino.bijedic at eracom-tech dot com): Sonicwall example VPN contribution.

第二章．硬件兼容性

2.1 支持的硬件架构

m0n0wall 只支持 X86 架构。所支持的设备包括标准的 PC 设备，以及各种嵌入设备。其目标是基于 X86 的嵌入 PC。

这意味着不包括非X86 的设备，如基于MIPS的Linksys设备，基于ARM的D-Link设备等。

FreeBSD 不支持 MIPS 和 ARM 平台。FreeBSD 所支持的平台可以阅 http://www.freebsd.org/platforms/index.html。其中列出的某些平台还不能工作，如MIPS。目前m0n0wall只支持X86 平台。

2.2 支持基于 PC 的硬件

m0n0wall 可以运行在任何标准 X86 PC，只要求具备两个网络接口。

2.2.1 最低要求

486 CPU - 任何 486 或更高的 CPU 就足够运行 m0n0wall。具体需要多高的 CPU 才满足需求，取决于多种因素，包括互联网的接入带宽，所需要的并发连接数，使用什么功能等。对于大部分应用，一个 486 或奔腾(Pentium)CPU 就足够了

64M 内存 - 64M RAM 是官方建议的最少配置。m0n0wall 的 CD 版本据报告在 32MB 内存机器中也运行得很好。当使用 m0n0wall 的 CF(Compact Flash)或硬盘版本，少于 64MB 内存时升级将失败，这是因为 m0n0wall 在 RAM 中保存所有东西，并没有使用交换空间(swap space) - 当内存耗尽，没有任何后援内存支持。

2.2.2 建议的 BIOS 修改

需要修改一些 BIOS 设备，才能让 m0n0wall 正确地工作。

Plug and Play OS

很多主板的 BIOS 都有一项“Plug and Play OS”设置其它类似的设置。这一项该设置为“NO”或“Disable”。通过关闭这些选项，让 BIOS 分配系统的资源而不留给操作系统来做。由 BIOS 负责分配资源，FreeBSD(包括 m0n0wall)运行得最好。

禁止不必要的设备

一般情况下不需要关心这一点，但如果遇到硬件相关的问题，我们建议在 BIOS 中禁止所有不必要的设备，如果主机板中的声卡，某种情况下的并口、串口，以及其它不使用的设备。如果不使用该设备，禁止它是安全的。

2.2.3 存储介质

m0n0wall 可以安装在 CF(compact Flash)，硬盘，或 CD，配一软驱保存配置。

CompactFlash

需要至少 8M Compact Flash 卡。

硬盘

一般的 IDE 或 SCSI 硬盘都可以（只要是 FreeBSD 支持的磁盘控制器）。

CD/软驱

任何 IDE 或 SCSI CD-ROM/DVD 驱动器都可以启动 m0n0wall。另外需要一个 1.44MB 的软驱（任何标准的软驱），一张用 MS-DOS/FAT 格式化后的软盘。使用这种配置方法，你的 PC 应该能够从 CD-ROM 启动。

Zip Drive setup

从版本 1.2b3 开始，m0n0wall 可以在一个 Zip Drive 中运行硬盘映象。使用写硬盘的方式来写 Zip Drive。

2.3 支持的嵌入式设备

以下的嵌入式设备可以运行 m0n0wall。

2.3.1. Soekris Engineering

所有 Soekris 设备都与 m0n0wall 完全兼容。对于 net4501 和其它 45xx 型号 ,使用 net45xx 映象。对于 net4801，使用 net48xx 映象。

规格

net4501-30: 133 Mhz CPU, 64 Mbyte SDRAM, 3 Ethernet, 2 Serial, CF socket, 1 Mini-PCI socket, 3.3V PCI connector.
net4511-30: 100 Mhz CPU, 64 Mbyte SDRAM, 2 Ethernet, 1 Serial, CF socket, 1 Mini-PCI socket, Single PC-Card socket, PoE
net4521-30: 133 Mhz CPU, 64 Mbyte SDRAM, 2 Ethernet, 1 Serial, CF socket, 1 Mini-PCI socket, Dual PC-Card socket, PoE
net4526-20: 100 Mhz CPU, 32 Mbyte SDRAM, 1 Ethernet, 1 Serial, 16 Mbyte CF Flash, 2 Mini-PCI sockets, PoE.
net4526-30: 133 Mhz CPU, 64 Mbyte SDRAM, 1 Ethernet, 1 Serial, 64 Mbyte CF Flash, 2 Mini-PCI sockets, PoE.
net4801-50: 266 Mhz CPU, 128 Mbyte SDRAM, 3 Ethernet, 2 serial, USB connector, CF socket, 44 pins IDE connector, 1 Mini-PCI socket, 3.3V PCI connector.

2.3.2. PC Engines WRAP

Wireless Router Application Platform (WRAP)

PC Engines WARP 主板与 m0n0wall 完全兼容。使用下载页面的 WARP 映象。

2.3.3. Nokia IPxxx boxes

Nokia IPxxx 盒子原来用于运行 CheckPoint，但它们作为标准的 PC 硬件，也可运行

m0n0wall。

你可以通过 eBay 约$100 买到 IP110 和 IP120。

IP110,120 和 130

三个 10/100 以太网接口

National GX 300Mhz CPU

110 有 64MB RAM，120 有 128MB，130 有 256MB

5GB 硬盘

两个串口（auxiliary and console）

IP330,440,530,650,740

即使在二手市场，这些盒子的价格都超出 m0n0wall 一般安装需要，组装一台标准的 PC 要便宜得多。但如果你手头有一个或可以买一个便宜的，它可以运行 m0n0wall。一些可选的接口不能工作，如 HSS1,T-1CSU/DSU, V.35 和 X.21 串口，OC-3 ATM, FDDI 等。但以太网可以正常工作。

注意：

使用Nokia硬件，需要一点小技巧，因为其NIC初始的MAC地址为FF.FF.FF.FF.FF.FF。完整的说明和图可以参阅：http://chrisbuechler.com/m0n0wall/nokia/ip110.html

2.3.4. NexCom NexGate Appliances

NexCom 的 Nexgate 产品线的规格都支持 m0n0wall。它们比 WRAP 和 Soekris 平台高端，因此也更加贵。有好几款配置供选择，基本型号价格从$500 起。

2.4 虚拟机

m0n0wall 能够很好地运行多种虚拟机中，如 VMware Workstation,GSX 和 ESX，以及 Microsoft Virtual PC 和 Virtual Server。

虽然这种配置可以工作，我们也不建议把任何防火产品运行在虚拟机中，虚拟机主要用在测试和开发环境。事实上，m0n0wall 的大部分文档就是 Chris Buechler 使用 10-15 台 VMware 虚拟机的环境下写出来的。

如果你计划使用虚拟机测试m0n0wall，我们建议使用Chris Buechler配置好的m0n0wall VMware 映象（http://chrisbuechler.com/index.php?id=18 ）。

在MS VPC或VS中使用m0n0wall，你可以在Chris Buechler的网站找m0n0wall images for Microsoft Virtual PC and Virtual Server来下载(http://chrisbuechler.com/index.php?id=31)，它由Chris Notingham制作的。

2.5 硬件选择

为 m0n0wall 选择最佳的硬件配置方案有实际的困难，因为网络环境是不断变化的。下面提供一些指引供选择满足要求的硬件。吞吐量在大多环境下都是基本的参数，再有就是可靠性和将来的扩展能力。

2.5.1 嵌入式设备

下面是一个粗糙的指引，可选择一款满足你应用环境的嵌入式设备，如果有的话。

2.5.1.1. Soekris 45xx

Soekris 45xx 可满足带宽小于 10Mbps 的互联网接入环境。如果使用 IPsec VPN ,一块 45xx 足以支持最大 3Mbps 的 IPsec 吞吐量。其它功能在性能上没有实质的差别。

如果你计划建立 DMZ 区或第二个 LAN 区，需要清楚接口间最大吞吐量。一块 45xx 接口间最大吞吐量约为 17Mbps，所以如果需要大于 17Mbps 的接口间吞吐能力，就应该选择更快的平台。

2.5.1.2. Soekris 48xx

Soekris 48xx 可满足带宽小于 30Mbps 的互联网接入环境。如果使用 IPsec VPN ,一块 45xx 足以支持最大 ?? Mbps 的 IPsec 吞吐量。

如果你计划建立 DMZ 区或第二个 LAN 区，需要清楚接口间最大吞吐量。一块 48xx 接口间最大吞吐量约为 40Mbps，所以如果需要大于 40Mbps 的接口间吞吐能力，就应该选择更快的平台。

2.5.1.3. WRAP

WARP 可满足带宽小于 30Mbps 的互联网接入环境。如果使用 IPsec VPN，一块 45xx 足以支持最大 ?? Mbps 的 IPsec 吞吐量。

如果你计划建立 DMZ 区或第二个 LAN 区，需要清楚接口间最大吞吐量。一块 WARP 接口间最大吞吐量约为 40Mbps，所以如果需要大于 40Mbps 的接口间吞吐能力，就应该选择更快的平台。

2.5.2 网卡

提示:

这适用基于 PC 的安装。

所选择的网卡（NIC）是一个影响性能的重要因素。性能差的网卡（NIC）会使用 CPU 忙于中断处理，使用 CPU 成为性能的瓶颈。性能好的网卡（NIC）可以增加最大吞吐量两倍到三倍，甚至更多。

FreeBSD 使用驱动名称后跟网卡序号来命名网卡设备 ,例如 ,你有两块 Intel Pro/100 网卡(驱动名 fxp)，一块 3COM 3C905 网卡(驱动名 xl)，相应的 FreeBSD 设备名称是 fxp0,fxp1 和 xl0。

Intel Pro/100 和 Pro/1000 在 m0n0wall 中显示最佳的性能和最稳定，一些差的网卡，如使用 Realtek 芯片(FreeBSD 驱动名 rl)这类 ,相比之下性能显得很差。如果你打算购买网卡来安装 m0n0wall，我们强烈建议购买 Intel 的网卡。你可以在 ebay 以不到$30 USD 买到 3-5 块(打包)。

在小吞吐量的环境，就象许多 6Mbps 或更小的宽带连接，任何网卡都足够。如果需要网卡间高速的吞吐能力（超过 30-40Mbps），满足多个 LAN 之间或 DMZ 和 LAN 之间的应用需求，使用高质量的网卡变得很重要。

2.5.3 CPU

一般情况下，CPU 是系统的瓶颈。差的网卡比好的网卡更消耗 CPU，所以选择 CPU 的一个主要因素就是所用网卡的质量。

如果使用一块性能好的网卡，如 Intel 的网卡，一般情况下测量线路速度，Pentinum CPU 可以达到 30-40Mbps，Pentium III 可达 100Mbps。而要达到 1000 兆的速度，需要一个 2.8+GHz Pentium 4。

2.5.4 内存

存放 m0n0wall 映象所需要的内存，任何情况下不超过 64M RAM。可以随意安装更多的内存，但即使启用所有功能，且负载很重，都不会耗尽 64MB。

2.5.5 存储介质

m0n0wall 可以很好地工作在硬盘、CF 卡（至少 8MB）。引导时，m0n0wall 被加载到 RAM 中，并且在 RAM 中运行。所以存储介质的速度和类型不是影响性能的因素。

慢速的存储介质，象 CF 卡，启动时会比硬盘用稍多一点的时间，选择存储介质只影响到启动时间。建议使用 CF 卡获得最大的可靠性，因为它比硬盘更少故障。

2.5.6 高吞吐量环境

几个接口间需要极高吞吐量的应用环境中，尤其在 1000 兆接口，必须要考虑 PCI 总线的速度。当系统中同时使用几块网卡，CPI 总线的速度很容易成为瓶颈。大部分主板只有一到两个 PCI 总线，而每一个最大带宽只有 133MBbps 或 1064Mbsp，小于 1000 兆网卡的传输速度。PCI-X 可以传输 1056Mbps，或约 8.25Gbsp。

如果支持 1000 兆的吞吐量，那么需要服务器级别的主板，带有 PCI-X 插槽和 PCI-X 网卡（NIC）。

2.6 无线网卡

在考虑把m0n0wall用作无线接入点前，请先阅读这个FAQ（http://doc.m0n0.ch/handbook-single/#FAQ.AP）。

这些卡被分成两个列表：常用的卡和停产/少用的卡列表。

2.6.1 不支持的卡

目前，所有b, b/g和a/b/g的无线网卡都与m0n0wall不兼容。这些设备的驱动只有FreeBSD 5.x 和 6.x 的版本，而m0n0wall 的BSD版本是 4.11。当m0nowall基于新版本的FreeBSD是，它们将会被支持（http://doc.m0n0.ch/handbook-single/#）。

2.6.2 常用的卡

下面的列表，在我们的认识范围内，是 100%准确的。如果发现不正确的地方，请向Chris Buechler 报告(mailto:cbuechler@gmail.com)。

并不是所有无线网卡都支持主机接入(hostap)模式!(如，可以作为一个接入点)。这是硬件本身的限制，不是 m0n0wall 或 FreeBSD。如果列表中没有注明“no hostap”，则表示支持。

> **提示：**
>
> m0n0wall 文档项目没有得任何厂商的认可，他们可以在 froogle.google.com 找到。我们这里的链接只是方便读者。除了兼容的硬件外，查找的结果中可能包含不相关的硬件。

- 3COM 3crwe737A AirConnect Wireless LAN PC Card
- Cisco Systems Aironet 340 - no hostap
- Cisco Systems Aironet 350 - no hostap
- Compaq WL100
- Compaq WL110
- D-Link DWL-520 - NOT DWL-520+ as it uses a different, unsupported, chipset.
- D-Link DWL-650 - Revisions A1-J3 ONLY. K1, L1, M, and P revisions not supported.
- Dell TrueMobile 1150 Series - no hostap
- Intel PRO/Wireless 2011 LAN PC Card
- Linksys Instant Wireless WPC11
- Netgear MA311
- Netgear MA401
- SMC 2632W PC Card
- SMC 2602W PCI
- US Robotics Wireless Card 2410
- NL-2511CD

miniPCI

- 2511MP
- Dell TrueMobile 1150 Series

2.6.3 停产/少用的卡

提示：

其中一部分不支持 hostap。了解它们是否支持，可在 Google 中查找网卡名和 FreeBSD，得到该卡使用的驱动。如果驱动是"wi"，则该卡支持 hostap。那些驱动不是"wi"的网卡，不支持 hostap。

- Accton airDirect WN3301
- Addtron AWA100
- Adtec ADLINK340APC
- Aironet 4500/4800 series (PCMCIA, PCI, and ISA adapters are all supported)
- Airway 802.11 Adapter
- Avaya Wireless PC Card
- BayStack 650 and 660
- Blue Concentric Circle CF Wireless LAN Model WL-379F
- BreezeNET PC-DS.11

- Buffalo WLI-CF-S11G
- Cabletron RoamAbout 802.11 DS
- Corega KK Wireless LAN PCC-11, PCCA-11, PCCB-11
- ELECOM Air@Hawk/LD-WL11/PCC
- ELSA AirLancer MC-11
- Farallon Skyline 11Mbps Wireless
- Farallon SkyLINE Wireless
- ICOM SL-1100
- Icom SL-200
- IBM High Rate Wireless LAN PC Card
- IO Data WN-B11/PCM
- Laneed Wireless card
- Lucent Technologies WaveLAN/IEEE 802.11 PCMCIA and ISA standard speed (2Mbps) and turbo speed (6Mbps) wireless network adapters and workalikes
- Lucent WaveLAN/IEEE 802.11
- Melco Airconnect WLI-PCM-S11, WLI-PCM-L11
- Melco WLI-PCM
- NCR WaveLAN/IEEE 802.11
- NEC Wireless Card CMZ-RT-WP
- NEC Aterm WL11C (PC-WL/11C)
- NEC PK-WL001
- NEL SSMagic
- Netwave AirSurfer Plus and AirSurfer Pro
- PLANEX GeoWave/GW-NS110
- Proxim Harmony, RangeLAN-DS
- Raytheon Raylink PC Card
- Sony PCWA-C100
- TDK LAK-CD011WL
- Toshiba Wireless LAN Card
- Webgear Aviator
- Webgear Aviator Pro
- Xircom Wireless Ethernet adapter (rebadged Aironet)
- ZoomAir 4000

2.7 以太网卡

m0n0wall 支持大多数以太网卡（NIC）。然而一部分相对更加稳定、少出问题且更快。一般来说，你会找到 m0n0wall 社团的看法是便宜的芯片，如 Realtek 芯片，相对质量好的芯片，如 Intel，会有更多问题和慢一些，不管运行什么操作系统或软件。如果运行一个高流量的防火墙，使用一块质量好的网卡尤其重要。在有负载的时候，便宜的网卡会让“中断”淹没你

的系统，因为中断会实质上占用大量 CPU 时间，使 CPU 成为防火墙的第一个瓶颈。高质量的网卡对于高吞吐量的环境非常重要。

相比之下，我个人推荐使用 Intel 的网卡。Intel Pro/100 很容易找到，购买也很便宜。你可以在 eBay 用不到$25 购买三块来装备你的防火墙。

2.7.1 支持的卡

我们建议尽管试试手中的网卡，只要不与兼容列表冲突。但对于新出的 1000 兆网卡，我们建议检查下面的兼容列表，或就用 Intel Pro/1000 网卡，它被支持得很好。

如果有任何关于网卡兼容的问题，参考 FreeBSD 411-RELEASE Hardware Notes(http://www.freebsd.org/releases/4.11R/hardware-i386.html#ETHERNET)中的网卡支持列表。

2.7.2 ISA 网卡

虽然大部分的 ISA 以太网卡都被支持，我们建议可能的情况下，都不要使用它们。它们非常耗费时间，且很难让它们正常工作。PCI 成本多一点点，在我看来，与它带来的头痛相比，还是很值的。只有一种情况应该使用 ISA 网卡，就是没有足够的 PCI 插槽了。

如果你有 ISA 网卡想试一试，意味着可能遇到各种问题。它可以不能与系统一起工作，尤其是与 PCI 网卡一起使用。如果你已经遇到过使用的问题，那已经是警告了。

如果你需要让 ISA 网卡工作起来，你可能需要修改一些设置。首先，大多 ISA 网卡，包括 3COM ISA 网卡，缺省都有一个“plug and play”模式。FreeBSD 并不总是与 plug and play

设备一起工作得很好。对 3COM ISA 网卡，在 3COM 的支持网站有一个 DOS 工具，需要在 DOS 中运行，用来手工修改网卡的各种资源。检查网卡厂商的支持网站，找关于禁止 Plug and Play 设置的信息，一般是关于网卡的跳线或一个固件修改工具。

需要做的另外一件事是修改系统的 BIOS。如你需要把被 ISA 网卡使用的 IRQ 设置为 ISA/PnP。

第三章. 安装

3.1 系统需求

m0n0wall 基于 X86 嵌入式 PC。来自 Soekris Engineering 的 net45xx/48xx 系列和来自 PC Engines 的 WARP 系列都是官方支持的 X86 嵌入式 PC。

m0n0wall也可以运行在大多标准PC，既可以把generic-pc映象写入一个小容量硬盘或CF卡，也可以使用CD-ROM+软驱的版本。由于m0n0wall基于FreeBSD4，所以大多数能用在FreeBSD的硬件，也同样可以用在m0n0wall。详细的硬件支持列表查阅FreeBSD/i386 Hardware Notes（http://www.freebsd.org/releases/4.9R/hardware-i386.html）。

运行 m0n0wall 所需内存建议是 64M。实际可能需要更少一些，尤其不是使用许多的功能/服务，但这不能保证固件更新功（m0n0wall 没有使用 swap 空间，当内存将要耗尽时，它不能做任何事情）。

3.2 获得软件

预先制作好的映象有用在 Soekris 的 net45xx/48xx 的版本，有用在 PC Engines 的 Wireless

Router Application Platform(WARP)的版本 ,用在标准 PC(包括嵌入式)的 CF/HD 的版本，还有一个用在标准 PC 的 CD-ROM(ISO)映象及其根文件系统的 TAR 包(tarball)。

下载你平台所需的软件，可以把你的浏览器指向：

http://www.m0n0.ch/wall/downloads.php

然后选择所需的下载链接。把文件下载到你的工作机器，你将要从机器把文件写入一个 CD-R 或一张 CF 卡，这在下一节说明。

3.3 安装软件

m0n0wall 被设计成从 CD 映象或 CF 卡/IDE 硬盘启动，下载所需的映象文件后，请准备好 CD 或 CF 卡。

3.3.1 制作引导 CD

你可使用一个 CD-ROM 一个软驱在标准 PC 中运行 m0n0wall。硬盘不是必须的 ,m0n0wall 可以从 CD 启动，然后在内存中开始运行。软驱用来保存配置信息。如果你希望在标准 PC 中使用 HD 而不 CD 运行 m0n0wall，请参阅下一节。

- 下载 ISO 映象，参阅 3.2 获得软件
- 把 ISO 文件刻录到一个 CD-R(或-RW)
  - FreeBSD(ATAPI Recorder)

    ```
    burncd -s max -e data cdrom-xxx.iso fixate
    ```

  - Linux(ATAPI w/ SCSI emulation)

    首先使用下面命令找出刻录设备的 SCSI ID/LUN：

    ```
    linuxbox# cdrecord --scanbus
    Cdrecord-Clone 2.01 (i686-pc-linux-gnu) Copyright (C) 1995-2004 J?rg Schilling
    Linux sg driver version: 3.1.25
    Using libscg version 'schily-0.8'.
    ```

```
scsibus0:
        0,0,0      100) 'LITE-ON ' 'COMBO LTC-48161H' 'KH0F' Removable CD-ROM
```

看到 SCSI ID/LUN 是 0,0,0。按以下方法进行刻录（使用你刻录机的最大速度替换 <max speed>）:

```
cdrecord --dev=0,0,0 --speed=<max speed> cdrom-xxx.iso
```

- 在 Windows 中，可以使用你喜欢的刻录程序（如 Nero）来刻录 ISO 映象 (2048bytes/sector, Mode-1)。

- 格式化一张标准的 1.44MB 软盘，使用 MS-DOS/FAT 文件系统
  - FreeBSD
    fdformat -f 1440 /dev/fd0 && newfs_msdos -L "m0n0wallcfg" -f 1440 /dev/fd0

    提示：如果软盘已经格式化(low-level)，fdformat 这一步可以省略。

  - Windows
    format A:

确认你的 m0n0wall PC 设置为从 CD-ROM 而不从软驱启动。

3.3.2 制作 CF 卡或 IDE 硬盘

你可以让 m0n0wall 运行一个使用 CF 卡作为主磁盘的系统中，如 Soekris 的盒子，或者一台配有 IDE 硬盘的标准 PC 中。m0n0wall 将从 CF 卡或硬盘引导，然后在内存中开始运行。它并不 SWAP 到 CF 卡或硬盘，也不写任何东西，除非你修改了配置并且保存配置。

- 下载所需的 CF/IDE 映象，参阅 3.2 获得软件

  把映象写入足够大的 CF 卡或硬盘（至少 5MB）。CF 卡或硬盘的额外空间被忽略，使用比映象大的 CF 卡或硬盘没有额外的益处。

  FreeBSD
  gzcat net45xx-xxx.img | dd of=/dev/rad[n] bs=16k

其中 n=CF 卡或硬盘设备号码(查看 dmesg)；net4801 使用 net48xx-xxx.img；WRAP 使用 wrap-xxx.img，带硬盘的标准 PC 使用 generic-pc-xxx.img。

忽略 Trailing garbage 的告警信息-这是由于数字签名引起的。

- Linux

gunzip -c net45xx-xxx.img | dd of=/dev/hdX bs=16k

其中 X=CF 卡或硬盘设备名称(用 hdparm -I /dev/hdX 查看)-一些适配卡，尤其是 USB，可能会作为 SCSI 设备/dev/sdX 来枚举。

忽略 Trailing garbage 的告警信息-这是由于数字签名引起的。

- Windows

physdiskwrite [-u] net45xx-xxx.img

其中 physdiskwrite 的版本为 0.3 或更新版，可以从 m0n0wall 的 physdiskwrite 页面(http://www.m0n0.ch/wall/physdiskwrite.php)找到。如果目标硬盘大于 800M，才使用 -u 标记(不包括方括号)，但切记选择了正确的硬盘！

为了确认你已经选择了正确的硬盘，请在插入 CF/HD 之前，先运行一次 physdiskwrite，并记下其输出：

```
physdiskwrite v0.5 by Manuel Kasper <mk@neon1.net>
Searching for physical drives...
Information for \\.\PhysicalDrive0:
          Windows:        cyl: 14593
                          tpc: 255
                          spt: 63
          C/H/S:          16383/16/63
          Model:          ST3120026A
          Serial number: 3JT1V2FS
          Firmware rev.: 3.06
```

现在你可以知道系统中当前所的驱动器，从而知道这些都不是想用的驱动器。记下型号和系列号。插入准备写入的 CF 卡或硬盘，然后再一次运行 physdiskwrite，你现在会看

到多一个驱动器输出信息，对比之命令的输出，就可以确定需要写入的驱动器。

3.3.3 其它安装方式

m0n0wall 的其它安装方式参阅其它文档的“安装”部分(http://doc.m0n0.ch/handbook-single/#OtherDoc.Installation)。

3.4 引导 m0n0wall

首次启动运行 m0n0wall，需要对它进行配置。一旦经过配置，下次启动将根据配置自动运行 m0n0wall。

当第一次引导 m0n0wall：

- 插入 m0n0wall 按照上面的方法制作的 CD、CF 或硬盘。如果使用 CD，还要插入一张格式化的空白软驱，确认软驱可以写入（没有写保护）及使用 FAT 文件系统格式化。
- 确认系统从 CD、CF 或硬盘启动。可能需要进行 BIOS 进行配置。
- 确认系统有控制台。对于 PC，确认键盘、显示器已经连接；对于 Soekris 盒子，串口就是控制台，把它连接到一个终端，或使用一条 NULL-MODEM 线，把它与另一台运行了终端仿真软件的 PC 连接。
- 对于 Soekris 或 WRAP 主板，确认控制台(串口)的速度在 BIOS 中设置为 9600bps（Soekris 盒子设置 ConSpeed=9600）。
- 把系统接上网络
- 引导系统，等待控制台菜单出现，然后分配网络接口，这在下面的章节说明。
- 使用 webGUI 完成 m0n0wall 的配置,这在下面章节说明。把配置保存到你的工作电脑，作为备份。

提示:

部分 Soekris net45xx 似乎有一个 BUG，有时一个字符会连续两次发送到串口控制台，第二个字符会被删除。这可以通过升级 BIOS 来解决（版本 1.15a 或更新）。

完成编辑配置以后，就可以准备运行了。你不需要重新启动 m0n0wall ，但你可能希望看到它按照配置直接启动到工作状态。

第四章．配置

当 m0n0wall 第一次引导，它使用缺省配置，即第一个网络接口作为 LAN 接口，IP 是 192.168.1.1，并充当 192.168.1.X 网段的 DHCP 服务器。大部分情况下，这个缺省配置足以让用户只需要连接 m0n0wall 的 LAN 接口，就能使用 LAN 中某台机器的浏览器连接到 192.168.1.1:80（WEB 服务器运行在 m0n0wall 盒子），然后就可以使用 webGUI 完成剩下的配置，具体配置方法在下面说明。一般情况下，第一次引导 m0n0wall 需要使用控制台设置（分配）网络接口，然后再使用 webGUI 完成剩余配置。网络接口也可通过 webGUI 进行分配，所以控制台菜单只是让用户可以访问 webGUI 的一个必要手段。

4.1 控制台菜单

在引导过程中，首先显示标准的 BIOS 信息和 FreeBSD 的引导信息，然后 m0n0wall 在控制台显示一个简单的菜单，而不显示登录提示。

使用控制台菜单，可以分配每一个网络接口的功能：LAN，WAN 或 OPT，OPT 表示额外的接口，如 DMZ 或无线接入点（Host Access Point）。在控制台只需要设置 LAN 接口，其它可以使用 webGUI 来设置。把 LAN 接口的 IP 改为 LAN 中某个 IP，这样就可以连接到

webGUI 进行其余的配置，这在下一节说明。

4.2 WEB GUI

把浏览器指向 m0n0wall 盒子可以对其进行配置，m0n0wall 在 LAN 接口运行一个 WEB 服务器，使用标准的 80 端口。第一次连接到 m0n0wall 的 WEB 服务器，它会提示输入用户名和密码。用户名使用 **admin**，缺省密码是 **mono**。为了提高安全性，可以在"常规设置"页面修改密码。

缺省配置也许已经满足使用要求，如果不是，可以查看下面说明的每一个页面，找到相关的项目进行修改。当完成修改并且已经保存了 m0n0wall 的配置，记住下载一份当前配置的备份到 LAN 中的其它机器。

当第一次访问 m0n0wall 的 webGUI，将会看到"系统状态"页面。所有页面的左边都有一个菜单，通过该菜单可以导航到其它页面。网络接口子菜单的条目可能与你的系统不一样，这与网络接口数目，以及接口命名有关。以下章节的组织形式与菜单的条目结构一样。

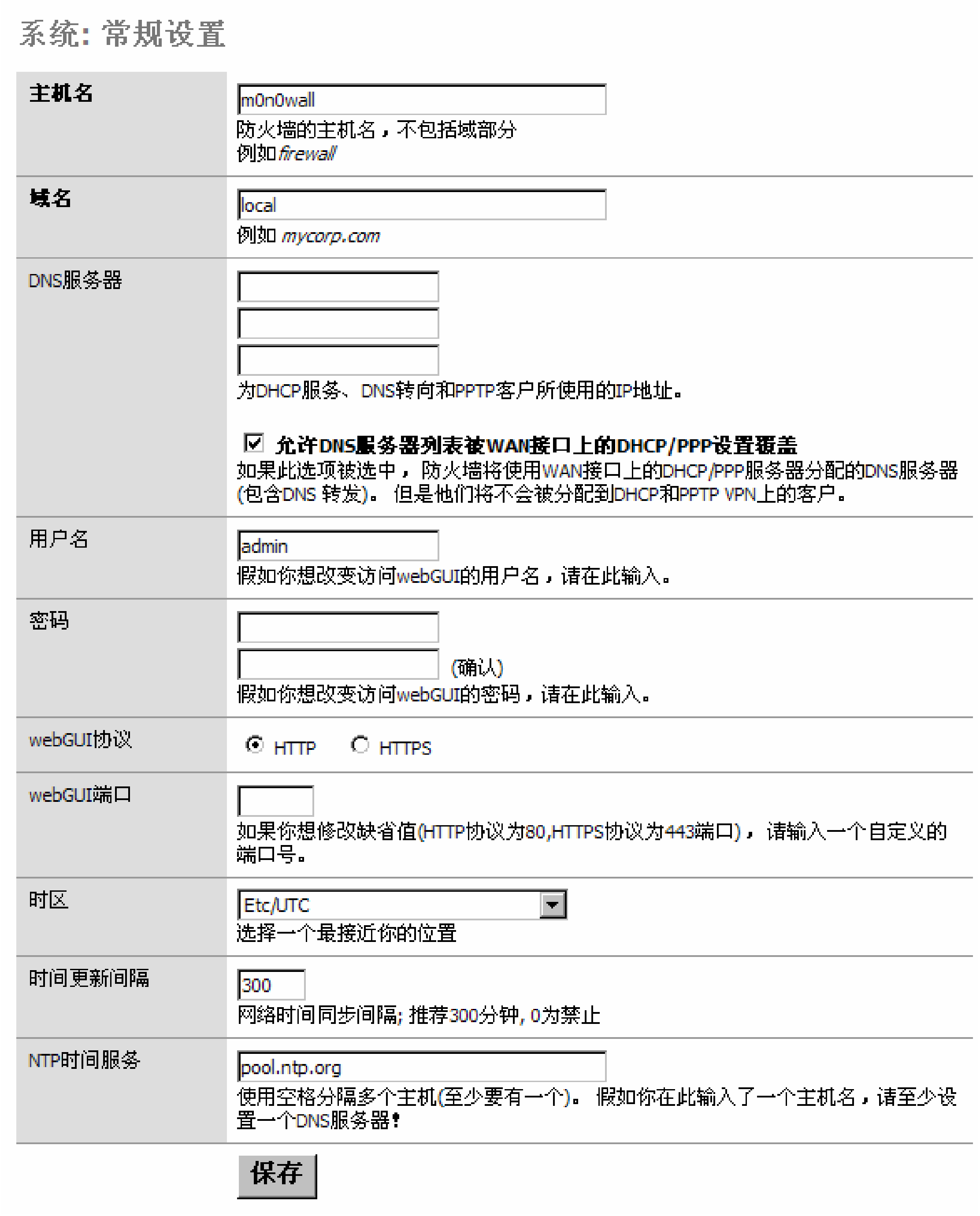
> **提示**
>
> 在以下章节中，部分截图包含模糊的区域。当查看你的 m0n0wall 的屏幕，那些都是系统相关的信息。

4.3 系统屏幕

4.3.1 常规设置

在常规设置屏幕可以为防火墙设置一些全局的参数。

图 4.1: 常规设置屏幕



在常规设置屏幕可以修改以下参数：

表 4.1. 常规设置参数

参数	说明	例子	参考
主机名	任意设定的防火墙主机名称	myfirewall	IP 基础
域名	防火墙主机所在域	Mydomain.com	IP 基础
DNS 服务器	防火墙用到的一个或多个 DNS 服务器的 IP。		DNS
用户名	连接 webGUI 的用户名	admin	
密码	连接 webGUI 的密码； 没有显示当前的密码，这一项仅用来改变密码。 首次安装 m0nowall，应该修改密码。		
webGUI 协议	m0n0wall 的 webGUI 所使用的协议。如果选择 HTTS，那么访问 webGUI 时要用“HTTS”开头。		
webGUI 端口	m0n0wall 的 webGUI 所使用的端口。如果不指定，则用缺省的 80 端口。		
时区	防火墙所在时区，这影响到日志里时间显示。		Logging
时间更新间隔	防火墙联系 NTP 更新本机时间的间隔。		Logging
NTP 时间服务器	防火墙所用的 NTP(Network Time Protocol)服务器名。		Logging

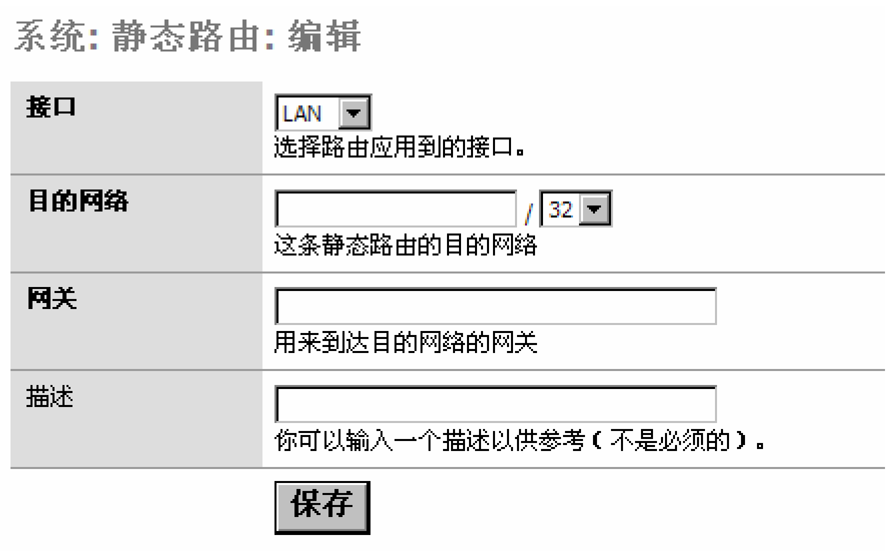
4.3.2 静态路由

对那些不能通过缺省路由到达的的网络，用户可以在这一个页面设置静态路由。点击“＋”图标，系统允许用户增加新的静态路由。

设置一条新路由的参数包括：

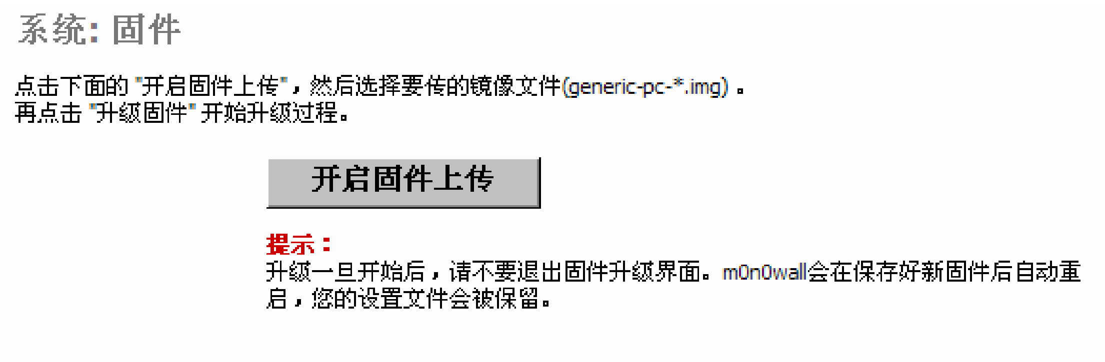
- 接口：选择新路由所在的网络接口（对应的网卡）
- 目标网络：这条路由要到达的目的网络，采用 CIDR 格式表示(Classless Inter-Domain Routing, RFC 1517,RFC1518,RFC1519,RFC1520)。
- 网关：用来到达目的网络的网关。
- 描述：你可以输入一个描述以供参考（不是必须的）。

图 4.3.2 新增静态路由屏幕



4.3.3 固件（升级）

图 4.3.3 固件屏幕



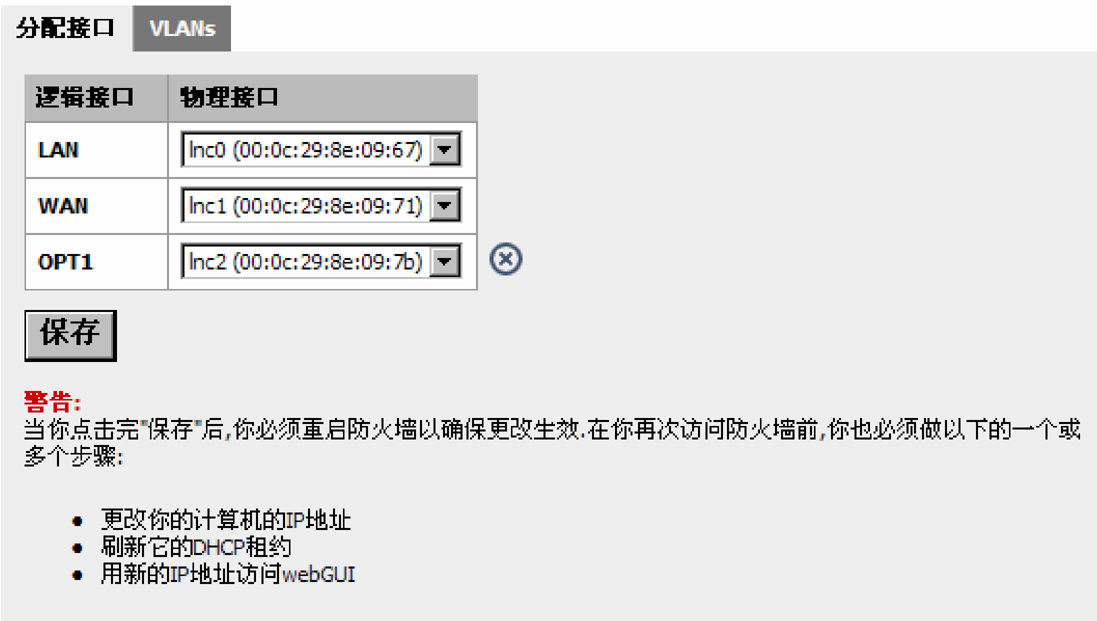
4.3.4 高级

[暂时略过]

4.4 接口屏幕

4.4.1 分配接口

进入“分配”子菜单可以为 LAN、WAN 分配系统中对应的物理接口。图：



还可以分配 VLAN。图：

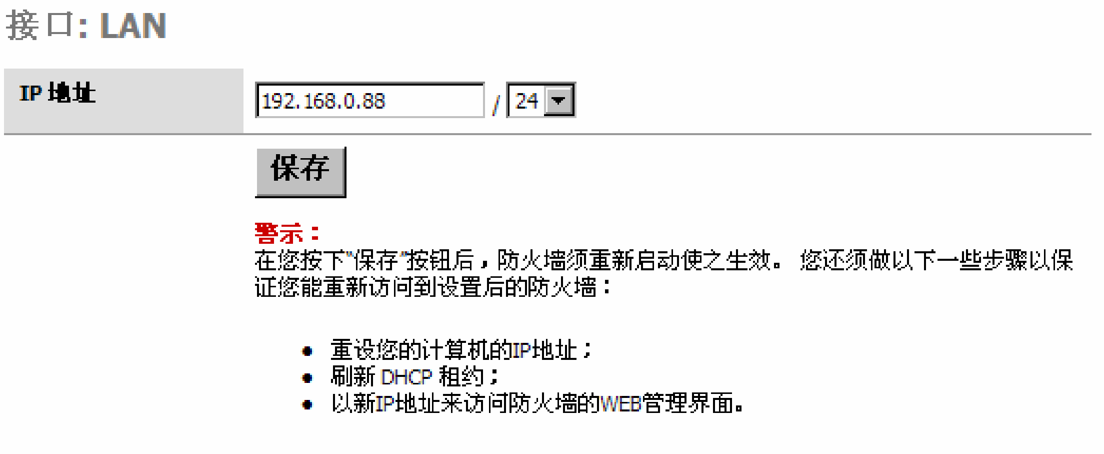


修改完成并保存后，需要重新启动系统才能生效。

4.4.2 LAN

在 LAN 页面可以修改连接内部网络的网络接口（网卡）的 IP 和子网掩码（CIDR 表示）。

完成修改并“按下”保存后，需要重新启动系统才能让修改生效。



4.4.3 WAN

在这一页面可以设置所有与 WAN 接口相关的参数。WAN 接口可以是静态 IP，一个 DHCP 地址，一个 PPPoE 接口，或者是一个 PPTP 连接。这些下面将说明。选择合适的 WAN 接口连接类型，同时填写连接类型对应的子表单。

下面详细说明每一项参数。

> ***提示***
>
> *当使用私有 IP 建立 Ipsec VPN 隧道时也不需要禁止 WAN 选项。对于进入 WAN 接口的 VPN 数据包，它们的源 IP 应该都是来自远端 VPN 设备的 WAN 接口，而不是来自远端私有网络。*

- 类型： WAN 的连接类型
  - 静态：为 WAN 接口分配一个静太 IP，同时指定掩码和网关。
  - DHCP：从 WAN 网络中的 DHCP 服务器分配一个动态的 IP
  - PPPoE：PPP over Ethernet，对于 ADSL 有用
  - PPTP：拨入 PPTP 服务器。有些 ADSL 服务商要求使用 PPTP 协议。
- 常规设置: 可以修改缺省的 MAC 地址和 MTU
  - 此处可以修改（“欺骗”）WAN 端接口的 MAC 地址（这在某些如有线宽带等情况下可能会用到），以 xx:xx:xx:xx:xx:xx 格式输入一个 MAC 地址或留空不作修改。
  - 按照 TCP 联接机制，MSS 值将等于您在本框输入的值减去 40（TCP / IP 头大小）。

若您在此处留空不填，PPPoE 连接的默认 MTU 值为 1492（8 个字节用于 PPPoE 的头和尾，各 4 字节），而其它连接类型则为 1500。

- 静态 IP 配置：如果 WAN 接口选择使用静态 IP，则这里静态 IP 和网关必须填写
  - IP 地址：设置静态 IP 及其掩码
  - 网关：填写缺省网关

- PPPoE 配置：如果 WAN 接口选择 PPPoE，这里设置 ADSL 拨号所需要的用户名其密码
  - 用户名：ADSL 服务商为你提供的名称
  - 密码：你的 ADSL 密码

- PPTP 配置：如果 WAN 接口选择 PPTP，这里设置建立 PPTP　ADSL 隧道所需要的参数。
  - 用户名：PPTP　(ADSL)服务商为你提供的名称
  - 密码：你的 PPTP　(ADSL)密码
  - 本地 IP：由服务商分配的本地 IP
  - 远端 IP：由服务商分配的远端 IP

- 阻止私有网络选项 :设置本项可以阻止来自 RFC1918 定义的私有网络((10/8, 172.16/12, 192.168/16，以及 127/8)的数据包。 这些数据包按理不应该出现在公网上， 它们的出现，如果不是因为网络上有设置不正确的路由器，就是恶意欺骗，所以打开本项是有益的。 当然，如果您的网络 WAN 端就是一个私有网，请不要打开它。

接口: WAN

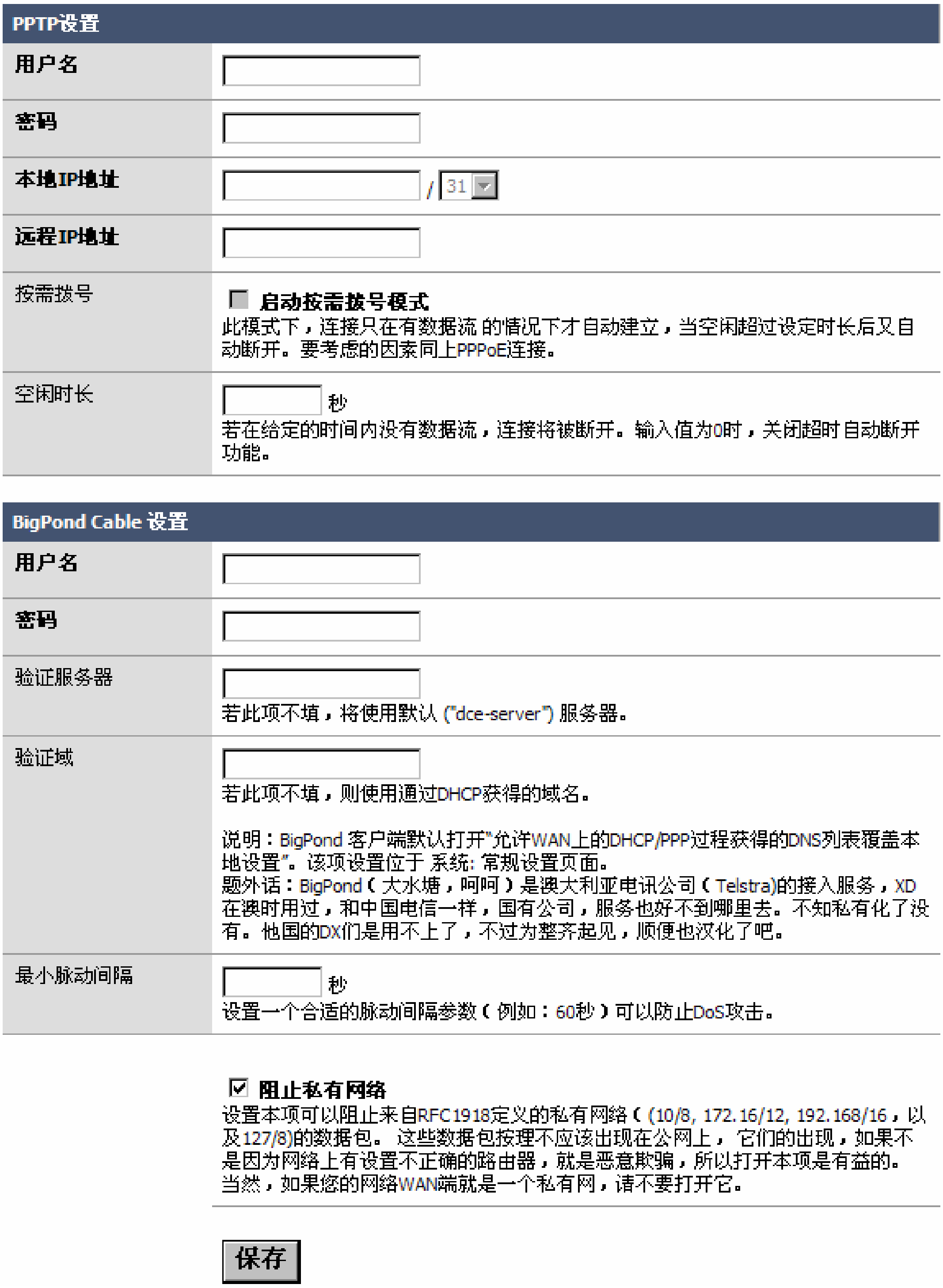
类型	DHCP

常规设置	
MAC 地址	此处可以修改（"欺骗"）WAN端接口的 MAC 地址 （这在某些如有线宽带等情况下可能会用到） 以 xx:xx:xx:xx:xx:xx 格式输入一个 MAC 地址或留空不作修改。
MTU	按照TCP联接机制，MSS值将等于您在本框输入的值减去40（TCP/ IP头大小）。 若您在此处留空不填，PPPoE连接的默认MTU值为1492，而其它连接类型则为1500。

静态IP设置	
IP地址	/ 31
网关	

DHCP客户端设置	
主机名	在发起一个DHCP请求时，这里输入的值会被发给DHCP服务器以作为用户确认和本机名。 一些ISP服务商会要求此项设置来用作用户确认。

PPPoE设置	
用户名	
密码	
服务名	提示：此处一般可不填。
按需拨号	☐ **打开按需拨号模式** 此模式主要用于计时用户，PPPoE连接只在有数据流 的情况下才自动建立，当空闲超过设定时长后又自动断开。 有两个因素要考虑，一是在连接建立前，系统要等待满足拨号条件， 再加上拨号时间（一般很短），所以有一些延迟。二是当局域网内 有主机染病毒及其它原因而不时自动向外网发数据的话，超时自动断开 的功能可能并不如您所预想的那样工作，因为在monowall看起来总有数据包要求通过。
空闲时长	秒 若在给定的时间内没有数据流，连接将被断开。输入值为0时，关闭超时自动断开功能。



4.4.4 可选的网络接口

可选的网卡可以用作各种目的。一般来说，它们被用作第二个 LAN 接口，或作为 DMZ 接口。

4.4.5 无线接口

4.5 服务菜单

4.5.1 DNS 转发

这个页面包含 m0n0wall 服务器 DNS 转发相关的选项。



启动 DNS 转发

选中“打开 DNS 转发功能”选择框，可以启动作用在 LAN 接口的 DNS 转发功能。要使这个功能，客户端机器要修改 DNS 配置，使用 m0n0wall 的 LAN 接口的 IP 作为它们 DNS 服务

器。

提示：

*在打开DNS转发器后，DHCP服务器（如果同时打开了的话 ）会自动地将LAN的IP地址作为DNS服务器设置给客户机，这样它们就可以用上DNS转发器了。 DNS转发器会使用在**系统：常规设置**中设置的DNS服务器作为转发对象 。又如果您勾选了“允许通过DHCP／PPP过程获得的DNS设置覆盖本地DNS列表” ,DNS转发器就会以获得的DNS服务器作为转发对象。 如果您没有勾选此项（或者您的WAN用的是静态IP），您就必须手工在 **系统：常规设置** 页面中设置好DNS服务器。*

注册 DNS 主机名

如果m0n0wall作为LAN的一个DHCP服务器，而且需要在LAN中进行主机名解析，可以选中“**在DNS转发器中注册DHCP租约中的主机名**”, 这样m0n0wall就会把***系统：常规设置***中的缺省域附加到主机名后，从而可以使用域名查找主机。如你的机器名称是my-pc，缺省域是example.com，m0n0wall会用动态分配的IP注册域名my-pc.example.com，其它主机可以使用该域名找到你的机器。

DNS 转发覆盖

如果需要对内部的主机(DNS 客户端)屏蔽某些域名，可以在这个页面把这些域名/IP 输入。如，你需要把 www.yourcompany.com 指向一台内部主机，而不是原来互联网中的那台，只需要在这输入该域名及对应的内部 IP 即可。这个功能可以作为简单的网站过滤器，用来限制客户端能访问的网站，做法是在这里给不希望客户端访问的网站域名分配一个非法 IP。如阻止访问 www.example.com，可以在这里输入一条覆盖记录，使该域名指向 1.2.3.4。虽

然这种方法可以很容易被绕过，如使用不同的 DNS 服务器或编辑主机文件(hosts)，但对于限制大部分用户对网站的访问还是足够的(配合防火墙，限制其它 DNS 请求出站)。

4.5.2 动态域名

动态域名可以让用户拥有一个固定的名称访问用户的网站，一般用在 IP 由 DHCP 服务器分配，而且经常改变的情况(如 ADSL)。动态域名可让用户使用动态公网 IP 建立 WEB 服务器，邮件服务器等。

浏览 m0n0wall 所用的动态域名客户端的网站 ez-ipupdate(http://www.ez-ipupdate.com/)，可以找到动态域名服务提供商的链接。

在继续配置之间，请先在动态域名服务商网站注册一个用户。

配置动态域名客户端

首先，选中“开启”选择框。

在“服务类型”下拉列表中选择你之前注册了的服务。

有些服务支持 MX DNS 记录(动态域名的一个子域名)，这可以确保收发邮件时能找到邮件服务器。如果你选择的动态服务支持这个功能(dyndns.org 就支持，其它大多也支持)，可以把邮件服务器的子域名填上。如果不需要 MX 记录，或服务商不支持这个功能，这一项留空。

泛域名(通配符)

如果需要使用泛域名，可以选中这一项。这意味着动态域名下的所有之域名，不管是否对其

进行配置，都将解析到你的服务器。如你的域名是 example.homeip.net，用泛域名时，example.homeip.net 下的所有子域名都解析到 example.homeip.net，如 www. example.homeip.net,mail example.homeip.net,等。

接下来的两项是动态域名用户名和登录密码。这是所选动态域名服务商的注册、登录信息。

点击“保存”,正常情况下 ,你的动态域名应该立即更新为当前 WAN 的 IP 地址。可以使用 PING 来确认 , 你应该得到 WAN 接口的 IP 地址。如果不是这样 , 到**诊断:系统**页面检看出错信息。



服务：动态DNS

动态DNS客户端	☑ 开启
服务类型	DynDNS
主机名	
服务器	填写要联接的服务器，一般毋需专门设置，故留空即可。
端口	填写要联接的服务器端口，一般留空即可。
MX	只在您需要特别MX纪录的情况下才设此项。不过，并不是所有动态域名服务商都支持本功能。
通配符	☐ 打开通配符支持。
用户名	
密码	

RFC2136 动态DNS更新	☐ 开启
主机名	
TTL	60 秒
键名	这里输入的键名须与提供RFC2136动态DNS服务的服务器上的设置相同。
密钥	请输入 HMAC-MD5 密钥，您可以将其粘贴过来。
协议	☐ 使用 TCP 而不是 UDP 协议

保存

说明：
为使动态域名更新能工作，您必须在系统：常规设置里设好DNS服务器或开启"允许通过DHCP/ PPP过程获得的DNS设置覆盖本地DNS列表"功能。

4.5.3 DHCP

可以在这一页面启动 DHCP 服务器，它可以运行在除 WAN 以外的所有接口。



启动 DHCP 服务器

选择一个接口(除 WAN 以外)页面，选中“在 xxx 端口上打开 DHCP 服务”。

拒绝未知客户

这一选项可以加强 DHCP 配置的安全性。许多公司都因为未经授权的机器接入其网络导致蠕虫病毒传播，这一选项可以保证只有经过授权的主机才可以从 DHCP 服务器分配 IP。当

启用这一选项时，只有在这里定义的主机才可以从 DHCP 服务器分配 IP(主机列表在本页面底部)。

这个方法的缺点是难以维护数量众多的主机 MAC，但很多情况下，我们发现增强的安全性相比所增加的工作量，是很值的。需要指出的是，这个方法只能阻止那些希望从 DHCP 服务器分配 IP 并且连接到互联网的人，任何有网络安全方面经验的人，可以很容易绕过这个限制。

对每一个选中的网络接口，可以独立设定 DHCP 服务的子网、子网掩码以及可用的 IP 范围。

范围

在第一个输入框输入开始地址，在第二个输入框输入结束地址。这个范围不应该与可用地址范围一样，因为可用地址范围包括了子网地址(网段 ID)、广播地址，还有被 m0n0wall 接口占用的地址。这些地址都不应该在所指定的范围内。

WINs 服务器

如果使用 NT 4 域控或 windows 2000 以前的客户端访问活动目录(AD)，需要在这里填入 WINs 服务器 IP。如果只有一个 WINs 服务器，可以留空第二个输入框。

DHCP 租约时间

如果 DHCP 客户端没有请求指定的租约时间，则使用缺省租约时间，其缺省值 7200 秒(两小时)。对于大部分网络环境，这个时长已经足够了。我一般建议把缺省时长设置为一周，即 604800 秒。

最大租约时间必须大于等于缺省租约时间。大部分网络根本不使用这个值。大部分情况下，

可把这个值设置为比缺省值多一秒。

点击“保存”然后点击“应用”, 就可以启动 DHCP 服务。

静态 DHCP 映射

通过静态 DHCP 映射，可以为同一台主机分配相同的 IP 地址。当基于 IP 地址定义访问规则，而仍然需要使用 DHCP，如防火墙规则或 LAN 中其它主机的访问规则，这个功能就能用上。

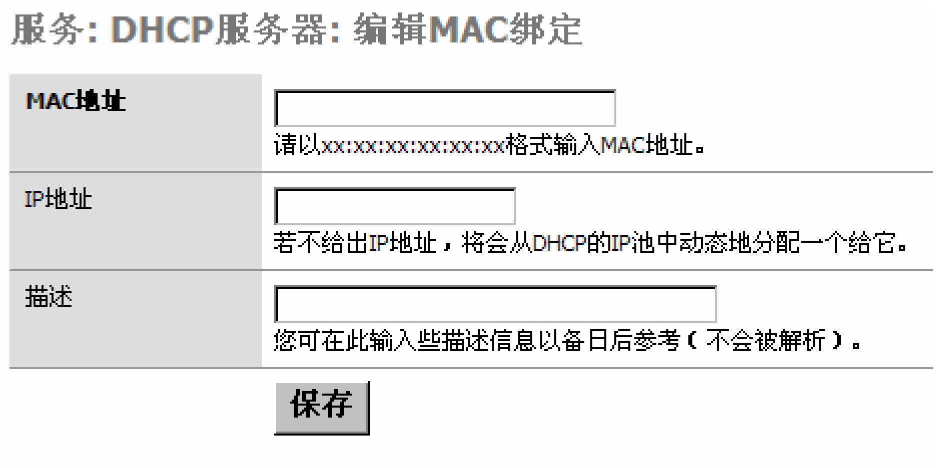
映射功能也可以作为“拒绝未知客户端”选项的一个例外，输入该客户端的 MAC，留空 IP 地址输入框，就可以从地址池中为该客户端分配一个 IP 地址。

在 MAC 地址框输入主机的 MAC 地址，格式为 xx:xx:xx:xx:xx:xx。在 Windows NT/2000/XP 客户端中，可以在命令窗口中运行“ipconfig”查看 MAC 地址。对于 Windows 95/98/ME 客户端，在开始->运行中运行 winipcfg。在 Unix 中可以运行 ifconfig 查看。

在 IP 地址框输入希望分配给客户商原 IP 地址，如果留空，则从可用 IP 地址池中分配动态一个 IP。如果指定一个 IP，则该 IP 不要求一定在可用 IP 地址池范围。

建议在说明框输入一些关于这个映射的说明，如目的、原因。

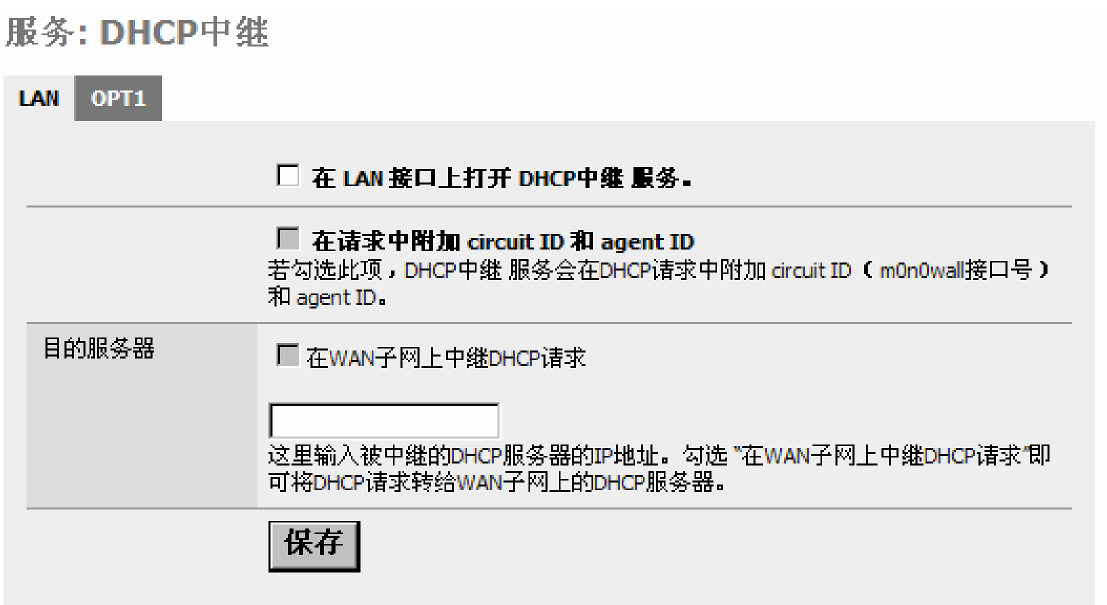
点击“保存”, 就可以保存映射设置。



提示

在常规设置中配置的 DNS 服务器(或 DNS 转发，如果启用的话)将会分配给 DHCP 客户。

4.5.3.1 DHCP 中继



4.5.4 SNMP

在这个页面，用户可以启动 SNMP 服务。如果用户有一个网络管理管理或监控系统，那么可以利用它们来管理 m0n0wall 设备。



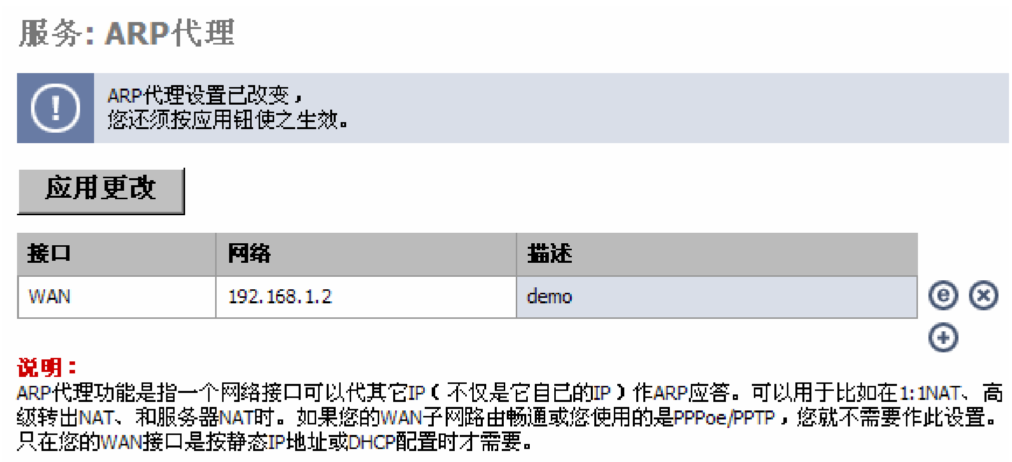
其中系统位置及系统联系人可以留空，但这些信息可以辅助你明确所监控的设备，尤其同时监控多台设备的情况。

SNMP 团体名称一般设置为 public，但如果有安全方面的考虑，那么应该把它设置成很难猜测到的名字，如字母和数字组合。SNMP 团体名称在网络中以明文方式传输，所以它可以被截获。但获得该名称的人最多可以了解到关于防火墙设备用途方面的信息(做不了更多事情)。在大部分环境下，这个问题可以不考虑，只需要清楚有这么个问题。

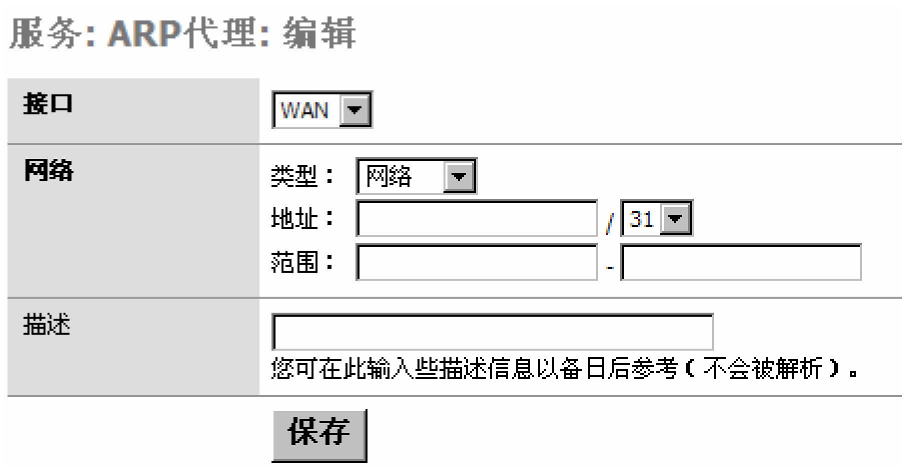
设置好希望的值后，点击“保存”以便参数起作用。

4.5.5 ARP 代理

ARP 代理功能是指一个网络接口可以代其它 IP（不仅是它自已的 IP）作 ARP 应答。可以用于比如在 1:1NAT、高级转出 NAT、和服务器 NAT 时。如果您的 WAN 子网路由畅通或您使用的是 PPPoe/PPTP，您就不需要作此设置。只在您的 WAN 接口是按静态 IP 地址或 DHCP 配置时才需要。



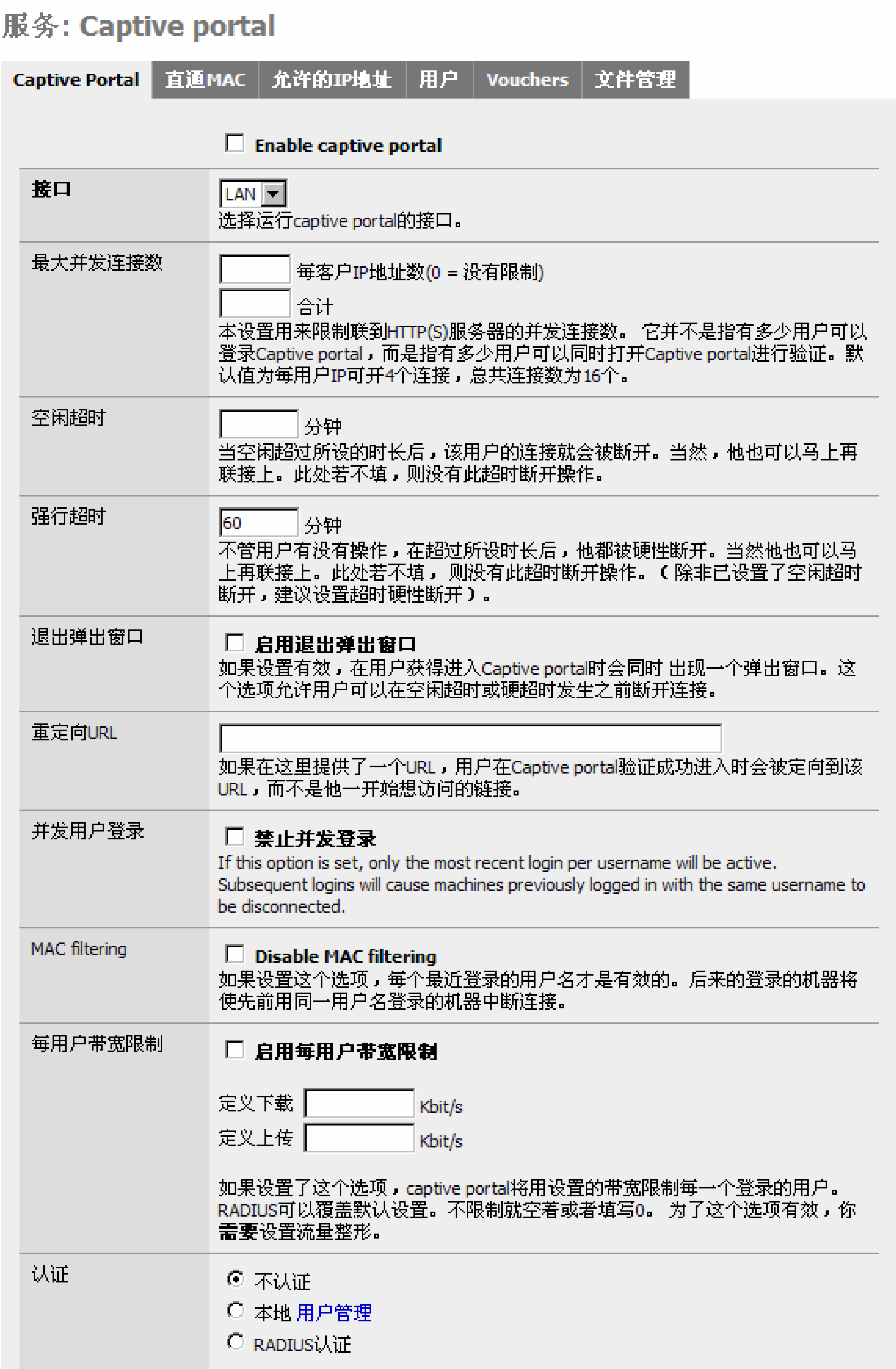
如果已经启用 1:1 NAT 或高级出站 NAT ,那么可能需要为那些进行 NAT 操作的 IP 启用 ARP 代理。增加代理记录，在这一页点击“+”按钮。



可以输入子网、单地址、范围三种地址类型。输入说明可以提醒用户这条记录的用途。点击“保存”，然后“应用更改”就可启用 ARP 代理。

更多关于什么情况下需要 ARP 代理及什么情况下不需要的说明，参考：http://doc.m0n0.ch/handbook-single/#Proxy.ARP。

4.5.6 Captive Portal



通过 Captive Portal 技术可以强制 HTTP 客户端打开一个特殊的网页（一般用于授权），在他正常浏览 internet 之前。其方法是拦截所有 HTTP 流量、不管其地访问的地址，直到用户允许从 Captive 退出（如通过授权）。

Captive Portal 技术经常用于无线(Wi-Fi)上网的环境，也可以用来控制有线上网的情况，如用在公寓、商务中心，还有给公众使用的“开放”以太网接口。

选中“启用 Captive Portral”即可启用此功能。

接口：选择 Captive Portral 功能所使用的网络接口（网卡），此功能只用使用其中一个网络接口。

最大并发连接数	本设置用来限制联到 HTTP(S)服务器的并发连接数。 它并不是指有多少用户可以登录 Captive portal，而是指有多少用户可以同时打开 Captive portal 进行验证。默认值为每用户 IP 可开 4 个连接，总共连接数为 16 个。
空闲超时	当空闲超过所设的时长后，该用户的连接就会被断开。当然，他也可以马上再联接上。缺省情况，没有此超时断开操作。
强行超时	不管用户有没有操作，在超过所设时长后，他都被硬性断开。当然他也可以马上再联接上。缺省情况， 没有此超时断开操作。（除非已设置了空闲超时断开，建议设置超时硬性断开）。

退出弹出窗口	在用户获得进入 Captive portal 时会同时 出现一个弹出窗口。这个选项允许用户可以在空闲超时或硬超时发生之前断开连接。
重定向 URL	如果在提供了一个 URL，用户经 Captive portal 验证成功后，会被定向到该 URL，而不是他一开始想访问的链接。
并发用户登录限制	如果设置这个选项，每个最近登录的用户名才是有效的。后来的登录的机器将使先前用同一用户名登录的机器中断连接。
禁止 MAC 过滤	如果设置这个选项，防火墙在用户登录后不再去验证其 MAC 地址是否保持不变。当用户的 MAC 地址不能被侦测（通常因为在用户和 Captive Portal 之间有路由器）的时候，这个选项就要被设置。
每用户带宽限制	如果设置了这个选项，captive portal将用设置的带宽限制每一个登录的用户。RADIUS可以覆盖默认设置。默认情况不限制。 为了这个选项有效，需要设置“流量整形”。 可以分别限制上传和下载带宽。
认证	支持本地认证和 Radius 论证，也可不认证。

HTTPS 登录	
HTTPS 服务器名	
入口页内容	可以为入口页上传一个HTML文件(提供缺省网页)。要求上传的页面须包含 (POST to "$PORTAL_ACTION$") 表单，再加一个提交按钮 (name="accept") 另加一个 name="redirurl" and value="$PORTAL_REDIRURL$" 的隐藏域。 包括 "auth_user" and "auth_pass" 及输入框（需要验证的话），否则验证不会成功。 表单示例： <form method="post" action="$PORTAL_ACTION$"> <input name="auth_user" type="text"> <input name="auth_pass" type="password"> <input name="redirurl" type="hidden" value="$PORTAL_REDIRURL$"> <input name="accept" type="submit" value="Continue"> </form>
认证错误页内容	当认证错误发生的时候，将显示这里上传的 HTML 文件的内容。 可以包括 "$PORTAL_MESSAGE$"， 它将会被替换为 RADIUS 服务器返回的错误信息。还可以在内面加入登录区，以供用户再次尝试。

4.5.7 LAN 唤醒

通过发送“Magic Pakcets”可以唤醒（开机）指定的服务器。前提是服务器的网卡支持 Wake On Lan 功能，并正确配置（WOL 接线，BIOS 设置）。

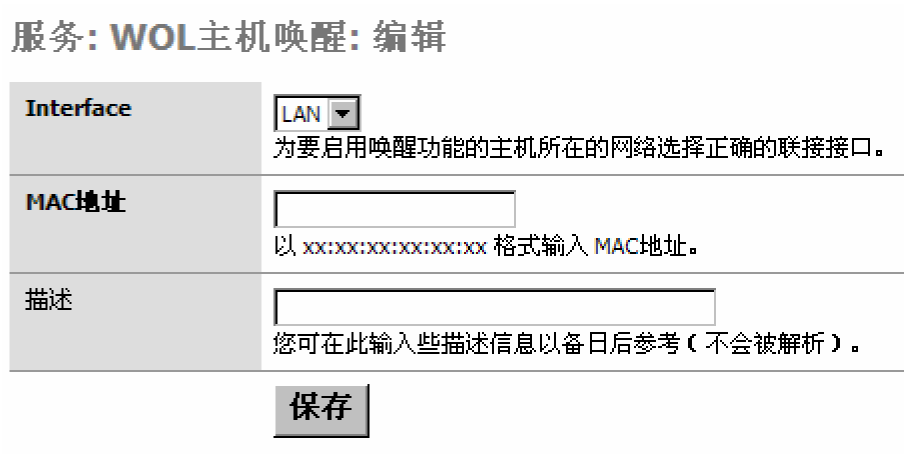
某种情况下，这个功能是有用的。如当你通过 VPN 连接家里或公司的网络，正好需要访问一台并不是总机的机器。这时可以登录与该机器在同一 LAN 的 Router/VPN 设备，然后该

机器发出一个 wake up 数据包。



通过唤醒功能开启某台机器，只需要输入其 MAC 地址，然后点击“发送”按钮。

如果需要经常使用这个功能，那么最好建立一份需要远程开机的机器列表。点击“＋”按钮，可以在以下列表中加入一台主机。对于已经加入到这个列表的主机，只需要点击其 MAC 地址，就可以远程开启它。



4.6 VPN 配置页面

略：在后面章节说明

4.6.1 IPsec

略

4.6.2 PPTP

略

4.6.3 PPTP 用户

略

4.7 系统状态页面

4.7.1 系统



4.7.2 网络接口

状态: 接口

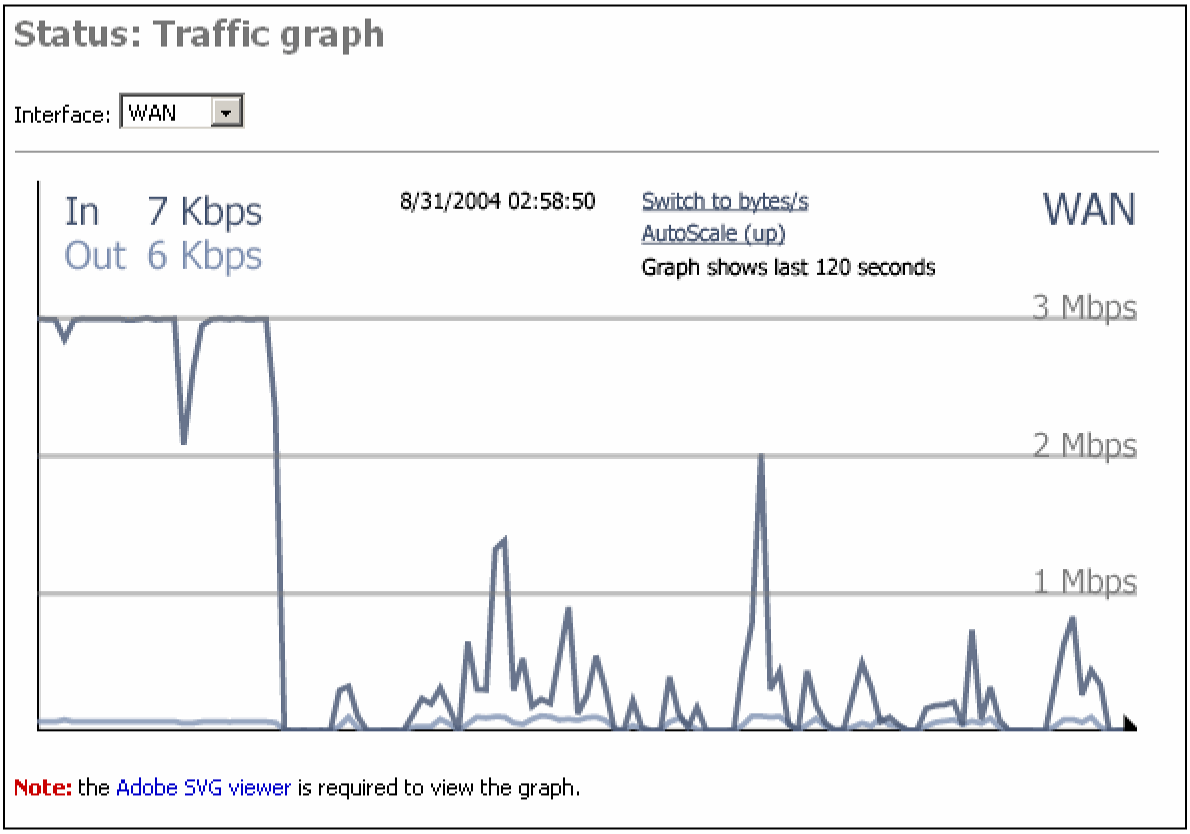
WAN interface	
状态	up
DHCP	up [illegible]
MAC地址	00:0c:29:13:70:7e
IP地址	192.168.5.168
子网掩码	255.255.255.0
网关	192.168.5.1
ISP DNS 服务器	202.96.134.133 202.96.134.188
入/出 包	42690/3 (2.51 MB/726 bytes)
入/出 错误	0/0
冲突	0

LAN interface	
状态	up
MAC地址	00:0c:29:13:70:74
IP地址	192.168.5.44
子网掩码	255.255.255.0
入/出 包	1851/42595 (218 KB/2.21 MB)
入/出 错误	0/1
冲突	0

注意：
如果任何包触发了它，按需拨号将再次的进行连接。为了证实这点：手工的断开连接**不能**防止按需拨号建立连接到外网！如果你想保持离线状态就不要使用按需拨号。

4.7.3 流量图

图 4.4: 流量图屏幕



流量图页面可以显示所选接口的实时流量，这一功能在版本 1.1 中首先引入。

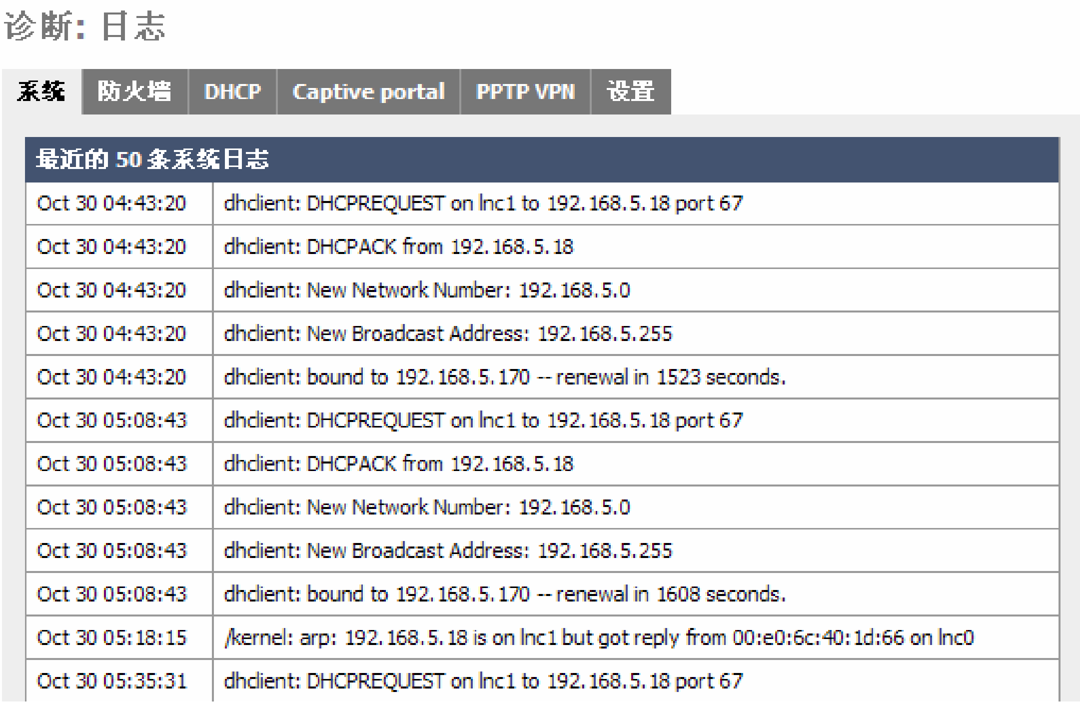
显示流量图需要安装 Adobe SVG Viewer 插件，这一页有一个连接指向该插件的安装页面。

4.7.4 无线

略

4.8 诊断页面

4.8.1 系统日志


诊断: 日志

系统 | 防火墙 | DHCP | Captive portal | PPTP VPN | 设置

最近的 50 条系统日志	
Oct 30 04:43:20	dhclient: DHCPREQUEST on lnc1 to 192.168.5.18 port 67
Oct 30 04:43:20	dhclient: DHCPACK from 192.168.5.18
Oct 30 04:43:20	dhclient: New Network Number: 192.168.5.0
Oct 30 04:43:20	dhclient: New Broadcast Address: 192.168.5.255
Oct 30 04:43:20	dhclient: bound to 192.168.5.170 -- renewal in 1523 seconds.
Oct 30 05:08:43	dhclient: DHCPREQUEST on lnc1 to 192.168.5.18 port 67
Oct 30 05:08:43	dhclient: DHCPACK from 192.168.5.18
Oct 30 05:08:43	dhclient: New Network Number: 192.168.5.0
Oct 30 05:08:43	dhclient: New Broadcast Address: 192.168.5.255
Oct 30 05:08:43	dhclient: bound to 192.168.5.170 -- renewal in 1608 seconds.
Oct 30 05:18:15	/kernel: arp: 192.168.5.18 is on lnc1 but got reply from 00:e0:6c:40:1d:66 on lnc0
Oct 30 05:35:31	dhclient: DHCPREQUEST on lnc1 to 192.168.5.18 port 67

4.8.2 DHCP 租约

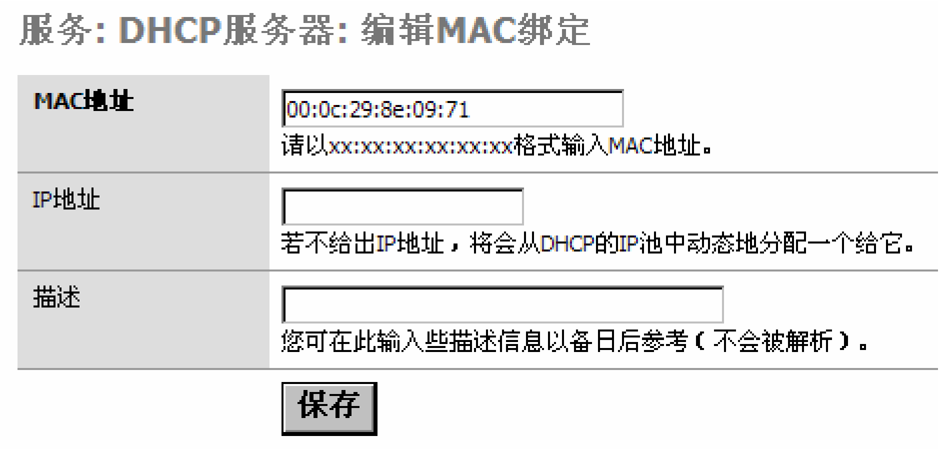

诊断: DHCP租约

IP地址	MAC地址	主机名	开始	结束
192.168.5.170	00:0c:29:8e:09:71		2007/11/24 22:31:38	2007/11/25 00:31:38

显示当前和过期租约

这个页面可以显示当前活动及过期的 DHCP 租约。点击“显示当前和过期租约”按钮，可以切换到只显示活动的租约。

点击“+”按钮，可进入 DHCP 绑定页面:

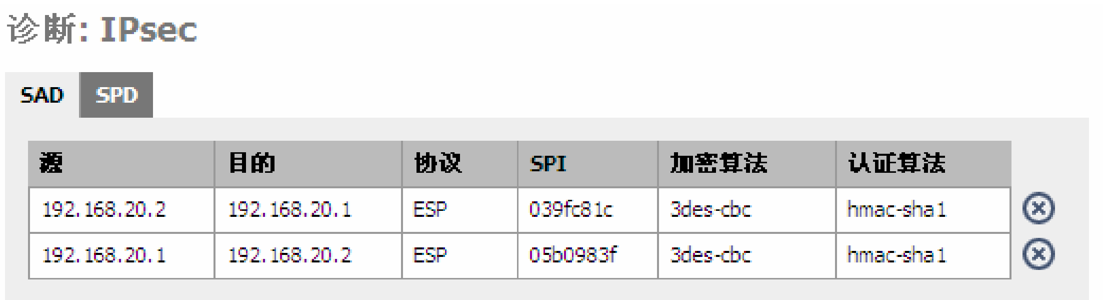


4.8.3 IPsec

IPsec 为两端的连接维护两个数据库：SAD 和 SPD。

安全联盟库（SAD）: Security Association Database

安全联盟库（SAD）维护当前所有安全联盟（SA’s）的列表。只有当 IPsec 连接建立后，才有相应的 SA。



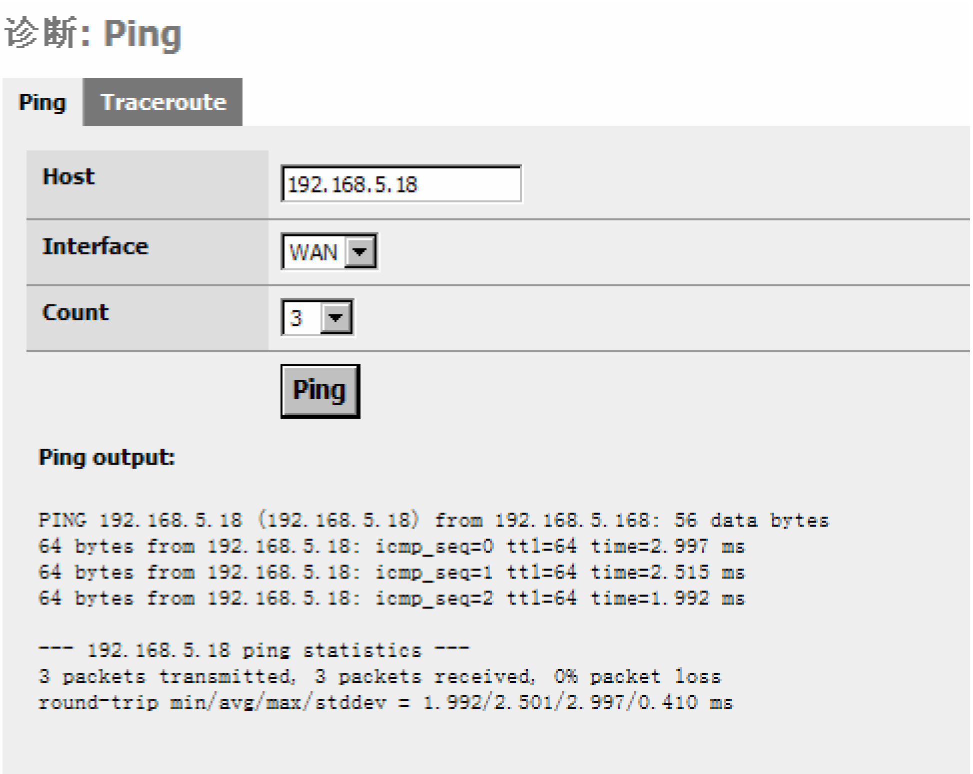
安全策略库（SPD）: Security Policy Database

安全策略库（SPD）维护系统中的 IPsec 策略列表。每一个 IPsec VPN 连接配置对应两条 SPD 记录，不管 IPsec 连接是否已经建立。系统根据这个库中的策略判断需要经 IPsec VPN 转发的 IP 流量。



4.8.4 SIP代理

4.8.5 ping/traceroute



这个页面提供了一个 GUI 的 ping/traceroute 界面。输入主机 IP，选择 PING 次数或路由的最大跳数，点击 ping/traceroute 按钮。

提示

这里提供的PING操作不能通过VPN发送PING包，基于同样的原因，SNMP也不能通过VPN连接。查看这里的FAQ｛http://doc.m0n0.ch/handbook-single/#FAQ.SNMPoverVPN｝获取更多相关的信息。所以不要使用这里的PING来判断VPN是否已经成功建立连接。

4.8.6 状态复位

这个页面可以复位 NAT 或防火墙的状态表。

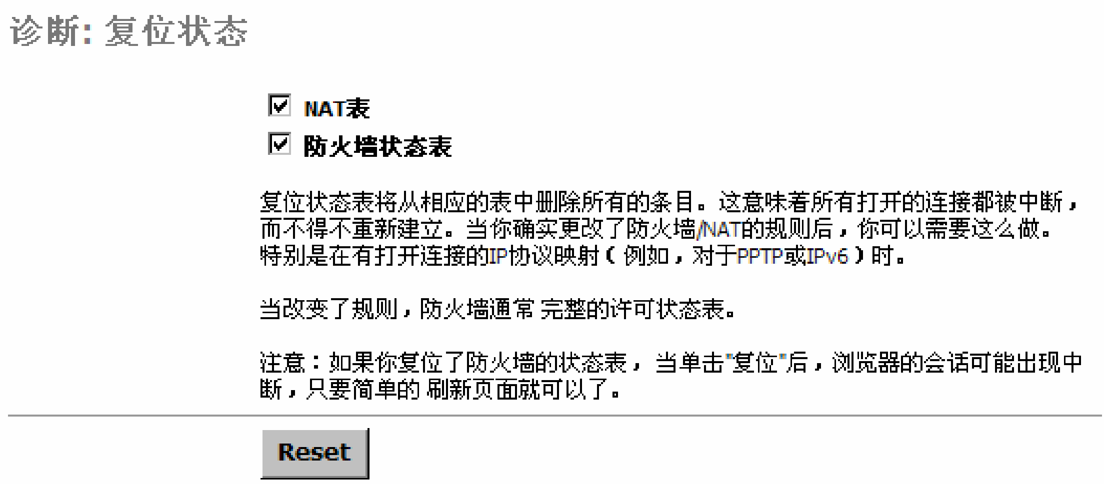
选择需要复位的状态，然后点击"复位"。

复位操作将删除相应状态表（NAT/Firewall）中的所有记录，这意味着将断开所有连接，连接需要重新建立。当对 NAT 或防火墙规则作了重要修改，需要进行复位操作，如涉及现存连接的 IP 协议映射（如 PPTP 或 Ipv6）。

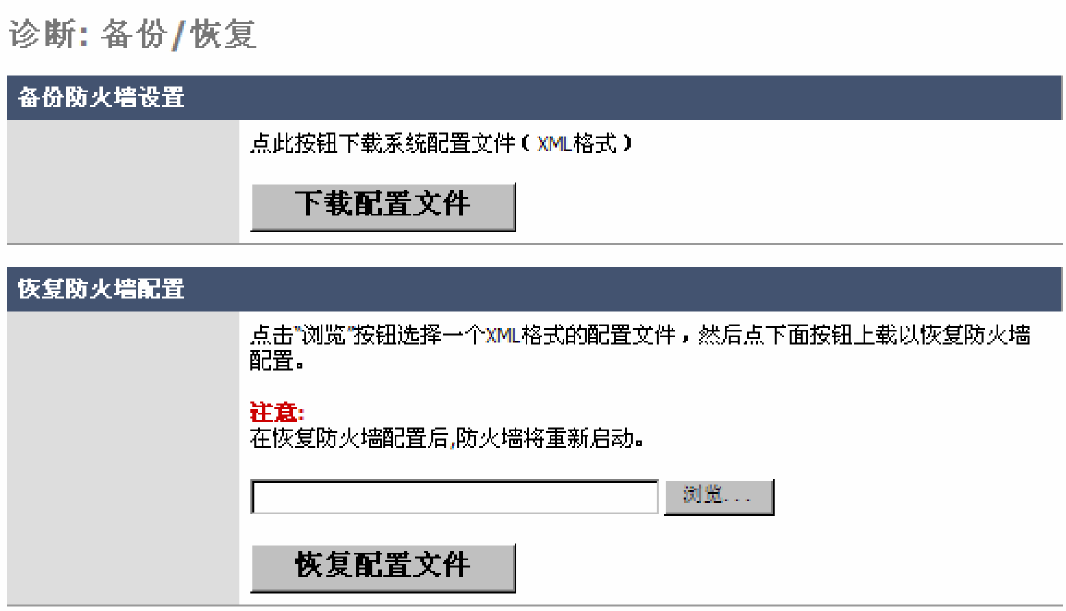
当修改规则时，防火墙通常不修改状态表。

提示

如果复位防火墙状态表，按下"复位"按钮后浏览器将会被挂起，这里只需要刷新一下页面。



4.8.7 备份恢复

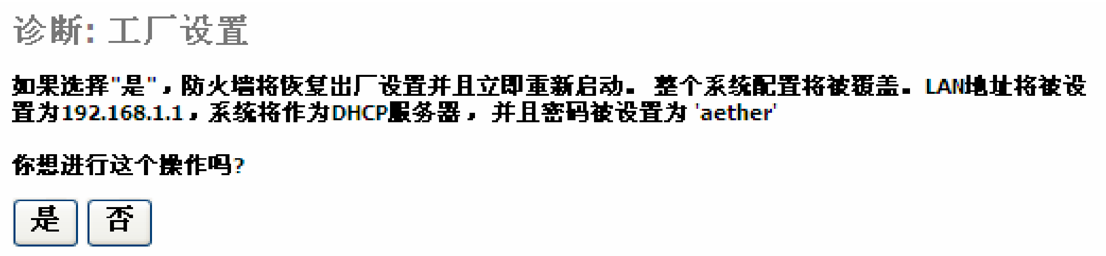


这个页面提供备份当前配置的操作，以及从备份文件恢复配置的操作。

点击“下载配置文件”，可以所当前配置保存到一个文件，通常是 config.xml。

当需要从一个备份文件恢复配置时，点击恢复防火墙配置栏目中的“浏览”按钮，选择之前备份的配置文件，然后点击“恢复配置文件”按钮。

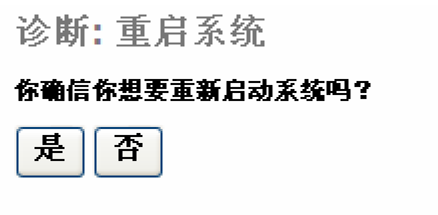
4.8.8 工厂设置



在这一页面点击"是", 将重置 m0n0wall 为缺省的配置，任何之前所作的配置都会被清除。

当各种配置尝试都失败，有时回到最初的配置可能是更容易的办法。这时可以使用这个功能把设备恢复到缺省配置。

4.8.9 重启系统



在这一页面点击"是"可以重启设备。

一般情况下，重启动并不能解决任何问题，但在很多情况下试一试是值得的。

与其很多系统不一样，重启动并不是所建议的维护步骤。除非有明确的原因，否则不需要重新启动设备。

第五章. 防火墙页面

5.1 规则

可以为每一个网络接口定义防火墙规则，如LAN，WAN，OPT1，OPT2…。缺省情况下，WAN接口将接拦截所有来自私有网络的IP包，如果你的WAN接口本身位于私有网络，则应该在接口->WAN中取消：**阻止私有网络**选项。

系统按照规则列表顺序进行规则匹配，如果设置了 BLOCK 规则，则应该特别注意其顺序。



5.1.1 操作（Action）

当规则匹配时，可以对 IP 包执行以下操作：

- Pass：让 IP 包通过。
- Block：丢弃 IP 包。

- Reject：丢弃IP包，向源IP发送TCP RST 或ICMP端口不可到达。只有当协议被设置为TCP或者UDP (但是不能是 "TCP/UDP")时Reject才是可用的。

5.1.2 禁用

设置这个选项是为了禁止这个规则（不起作用），但是不从列表中删除。

5.1.3 接口

选择防火墙规则作用到的网络接口。这条规则只过滤经过所选接口的 IP 包。

5.1.4 协议

选择匹配协议：

- TCP
- UDP
- TCP/UDP
- ICMP
- ESP
- AH
- GRE
- IPv6
- IGMP
- Any(表示任何协议)

5.1.5 ICMP 类型

当协议选择 ICMP 时，可以进一步选择 ICMP 的类型：

- Destination unreachable
- Echo
- Echo reply
- Source quench
- Redirect
- Time exceeded
- Parameter problem

- Timestamp
- Timestamp reply
- Information request
- Information reply
- Address mask request
- Address mask reply

5.1.6 源(source)

定义 IP 包的来源，包括：

- 任意
- 单个主机或别名
- 网络
- WAN 地址
- LAN 子网
- PPTP 客户
- OPTn 子网(表示可选网络接口所在的子网)

反转：表示规则需要匹配不是我们定义的来源（取反操作）。

5.1.7 源端口范围

为这条规则指定数据包的源端口或端口范围。通常不等于目的端口范围(而且经常被设置成"any")。 如果你想过滤单个端口，你可以让 '*到*' 字段空白。

5.1.8 目的（Destination）

定义 IP 包的目的地址，包括：

- 任意

- 单个主机或别名
- 网络
- WAN 地址
- LAN 子网
- PPTP 客户
- OPTn 子网(表示可选网络接口所在的子网)

反转：表示规则需要匹配不是我们定义的目的址（取反操作）。

5.1.9 目的端口范围

为这条规则指定数据包的目的端口或端口范围。通常不等于源端口范围(而且经常被设置成"any")。 如果你想过滤单个端口，你可以让 '*到*' 字段空白。

5.1.10 碎片

这个选项增加防火墙上的负载，同时使防火墙容易遭到DoS攻击。 在绝大数情况下，它是不需要的。如果你连接某些站点有问题试者让它有效。

5.1.11 日志

日志开启选项：当IP包匹配规则时记录日志。

防火墙被缺省时使用本地日志空间。不要为每个事件都开启日志。如果想做更多的日志，可以考虑使用远程syslog服务器 (参看 诊断：系统日志：设置 页面)

5.1.12 说明

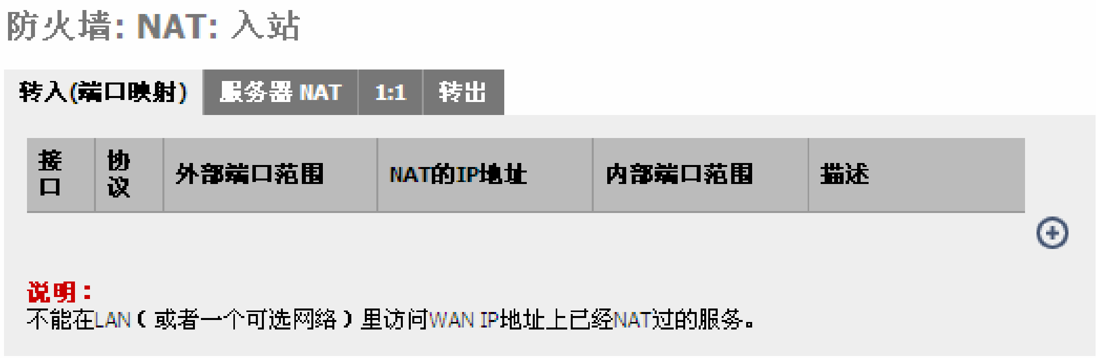
为了便于管理，最好为每一条规则填写一些说明。

防火墙: 规则: 编辑

Action	Pass 选择匹配的数据包按照指定的标准去做 提示：block和reject之间是不同在于reject返回给发送者一个数据包（TCP RST或者ICMP端口不可达），而block将默默地丢弃数据包。两种情况，最初的数据包都被丢弃。只有当协议被设置为TCP或者UDP (但是不能是 "TCP/UDP")是Reject才是可用的。
禁用	**禁用这个规则** 设置这个选项是为了禁止这个规则但是不从列表中删除。
接口	WAN 选择一个包一定会经过来匹配这个规则的接口。
协议	ICMP 选择规则要匹配哪种IP协议。 提示： 在绝大多数情况下，这里都指定*TCP* here.
ICMP 类型	any 如果你为上面的协议选择了ICMP，那么你可以指定一个ICMP类型。
Source	**反转** 使用这个选项反转匹配的规则。 Type: OPT1 subnet 地址: /
源端口范围	from: any to: any 为这条规则指定的源数据包端口 或端口范围。通常不等于目的端口范围(而且经常被设置成"any")。 提示： 如果你想过滤单个端口，你可以让 *'到'* 字段空白。
Destination	**反转** 使用这个选项反转匹配的规则。 类型: PPTP客户 地址: /
目的端口范围	from: any 到: any 为这条规则指定的目的数据包端口 或端口范围。通常不等于目的端口范围。 提示： 如果你想过滤单个端口，你可以让 *'到'* 字段空白。
Fragments	**允许碎片包** 提示： 这个选项在防火墙上 发出附加负载，同时是的防火墙容易遭到DoS攻击。在绝大数情况下，它是不需要的。如果你连接某些站点有问题 试着让它有效。
Log	**记录被规则捕获的数据包** 提示： 防火墙被限制为本地 日志空间。不要为每个事件都开启日志。如果想做更多的 日志，可以考虑使用远程syslog服务器 (参看 诊断：系统日志：设置 页面)。
描述	为了参考你可以输入一个描述（不是必须的）。
	保存

5.2 转入 NAT（端口映射）

转入 NAT（端口映射）可以把公网 IP 的端口映射到 LAN 或 OPT（可选）网络。



点击“+”图标，可以增加新的映射规则：



5.2.1 接口

一般情况下选择 WAN，以便允许源自 Internet 的流量可以进入。也可以选择 OPT（可选）网络接口（如果多于两块网卡或分配了 VLAN）。

OPT（可选）接口可以作为 DMZ 接口使用，以便允许经 DMZ 访问 LAN 中某主机的端口。例如，你需要使用一个位于中 LAN 中的 DNS 服务器，可以设置一条映射 UDP Port 53 到 LAN DNS 服务器 IP 的 NAT 规则，这样就可以把 DMZ 接口的 IP 当作 DNS 服务器的 IP 使用。(这个例子中，使用 NAT 方式并没有比使用防火墙规则允许 IP 包经 DMZ 直接到达 LAN DNS 服务器来得方便)。

5.2.2 外部地址

一般情况下，外部地址就是 NAT 接口（WAN 或 OPT）的 IP。但如果你的 WAN 或 OPT 接口有多个公网 IP，也可选择其它公网 IP，但必须首先在“服务器 NAT”页面定义，如图：



5.2.3 协议

进行映射的协议，支持：TCP，UDP，或 TCP/UDP。

5.2.4 外部端口范围

可以从下拉列表中选择协议，或输入端口范围。若您只想作一个单端口映射，可将 *'到:'* 框留空。（注：如果这里指定一个范围，则本地端口也将对应一个范围，但只需要指定起始端口）

5.2.5 NAT IP

指内部服务器的IP，一般是LAN中某主机的IP。例如： *192.168.1.12。*

5.2.6 本地端口

为上面选定的主机IP地址指定端口号。若是一个端口范围， 只需指定起始的端口号（终止端口号会自动算出）。

提示：这里的值通常与上面外部端口号中的 “从：”区相同。

5.2.7 说明

为了便于管理，你可以输入一个描述/说明 （不是必须的）。

5.2.8 在防火墙中自动添加一条允许NAT规则通过的过滤规则

建议让系统自动添加一条规则。

点击“保存”, 结果如图：



接口	协议	外部端口范围	NAT的IP地址	内部端口范围	描述
WAN	TCP	23 (Telnet)	192.168.16.101	23 (Telnet)	test
WAN	TCP/UDP	21 (FTP)	192.168.16.100	21 (FTP)	test

5.2.9 编辑转入（端口映射）的防火墙规则

当添加了一条新的NAT转入规则（端口映射规则），并且允许系统自动添加一条允许NAT规则通过的防火墙过滤规则。这时你可能希望查看和编辑该防火墙规则。例如你可能不允许整个internet访问你开放的服务。

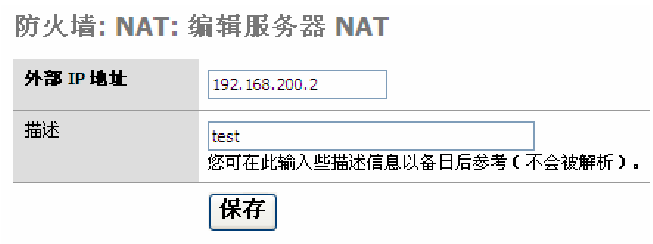
编辑可以到页面：防火墙->规则。

5.3 服务器 NAT

如果用作 NAT 的接口有多个公网 IP，可以先在这里定义，然后定义转入 NAT 规则时就可以选用。

5.3.1 增加服务器 NAT 记录

防火墙->NAT->服务器 NAT，点击“+”图标：

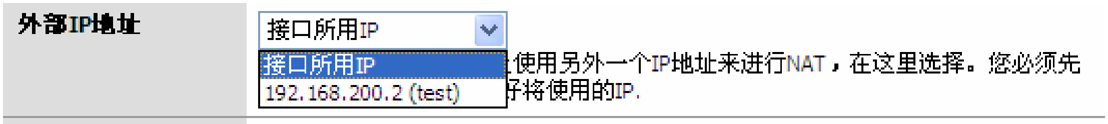


点击保存后，结果如图：



5.3.2 使用服务器 NAT 记录

现在进入转入 NAT（端口映射）页面，点击“+”增加一条新的规则，这时，外部地址的下拉列表多了“服务器 NAT”所定义的 IP。



5.3.3 启动代理 ARP

根据 WAN 接口的连接设置，为了使 NAT 工作，可能还需要设置代理 ARP 服务。

如果满足以下条件，就不需要设置代理 ARP：

- WAN 接口的额外 IP 可以经由 ISP 正常路由到你的设备。
- WAN 接口使用 PPPoE 或 PPTP 建立连接。

以上情况使用代理 ARP 不起什么作用。但如果 WAN 使用静态 IP 或 DHCP，而且没有正确的路由到达 WAN 接口所分配的 IP，这时，在服务->ARP 代理 中增加相应的记录，可确保 WAN 接口为与它绑定的 IP 响应 ARP 查询。

5.4. 1:1NAT

1:1 NAT 实现内部 IP 与外部 IP 之间的映射(与端口无关),一般是内部私有 IP 与外部公网 IP 之间的映射.当设置一条 1:1NAT 规则,所有源自指定内部 IP 的流量都会被 NAT 到指定的外部 IP(位于 Internet),同样,所有源自指定外部 IP 的流量,都会被 NAT 到指定的内部 IP.(缺省情况下,防火墙规则不允许任何转入流量到达 1:1NAT 映射指定的内部主机)

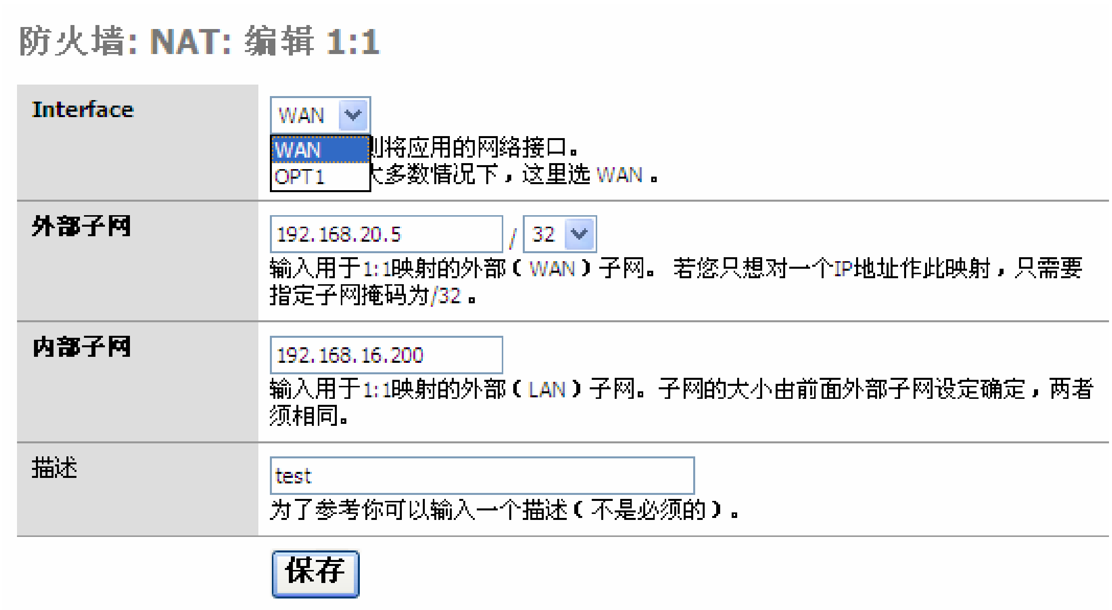
1:1NAT 可以映射一个子网.

1:1NAT 可以使用 OPT(可选)网络接口,但一般不使用.



5.4.1 添加 1:1NAT 记录

进入防火墙: NAT: 1:1 页面,点击“+”可以添加新规则：



5.4.1.1 接口

一般情况下都选择 WAN，OPT 接口也可以选择。

5.4.1.2 外部子网

外部子网可以设定希望映射的 IP 或子网。一般情况下映射一个 IP（子网掩码为/32），但如果你拥有一个段的公网 IP，如某一个 C 段地址，而且你的 LAN 或 DMZ 也是一个 C 段地址，而你希望实现内外两个网段之间的 1:1NAT 映射，这时可以在这里指定一个子网。

5.4.1.3 内部子网

大多数情况下，这里指定一个 IP（位于 LAN 或 DMZ 中）。对于映射一个子网的情况，这里输入子网 IP，其掩码与外部子网相同（1:1NAT 要求两者必须相同）。

5.4.1.4 说明

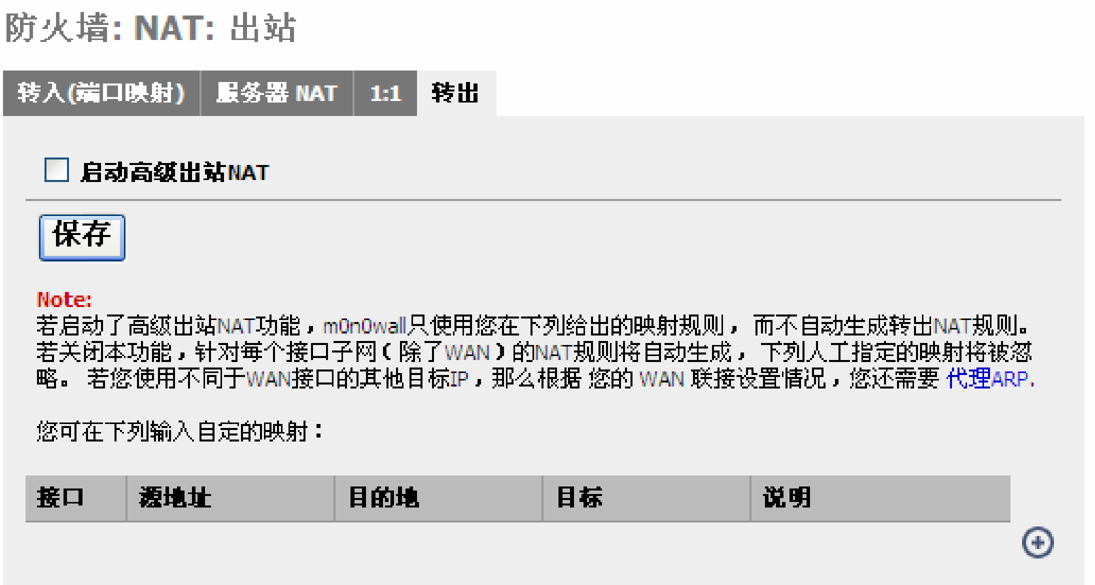
这里的说明不是必须的，但为了便于管理，建议输入。

输入所设置后，点击“保存”，然后“应用”所作的修改。

5.5 转出 NAT

缺省情况下，转出 NAT 功能是自动完成的，如 LAN 中的主机通过我们的设备共享访问互联网（上网）。

但也可以人工指定映射规则，这里需要启动“高级出站 NAT”功能。



若启动了高级出站NAT功能 ,m0n0wall只使用人工设定的映射规则，而不自动生成转出NAT规则。 若关闭本功能，针对每个接口子网（除了WAN）的NAT规则将自动生成，人工指定的映射规则将被忽略。 若您使用不同于WAN接口的其他目标IP，那么根据您的 WAN 联接设置情况，您还需要 代理ARP.

5.5.1 添加转出 NAT 规则



5.6 流量整形

流量整形功能可以根据协议、端口等参数为各种流量设定其占用的最大带宽。

使用流量整形的基本方法：手工配置规则，或者使用“整形向导”。

- 手工配置规则：按以下步骤进行
  - 定义“管道”: 定义带宽分配/限制的参数（属性）
  - 定义“规则”: 定义流量相关的参数（属性），这里需要引用“管道”，把“管道”所定义的带宽分配属性应用到相关的流量，达到带宽限制的目的。

- 使用“整形向导”: 由系统根据上行、下行带宽，自动生成“管道”、“规则”。在此基础上，用户可以编辑自动生成的规则。



5.6.1 添加管道



点击“+”按钮，可以添加新的管道：

防火墙: 流量整形: 编辑管道

带宽	1946 Kbit/s
延迟	ms 提示： 绝大多数情况下，这个字段指定为0（或者空着）。
丢包率	提示： 绝大多数情况下，这个字段指定为0（或者空着）。0.001的意思是1000个包丢弃1个。
队列大小	通道 提示： 在绝大多数情况下，你会让这个字段空着。 所有在这个管道里的包首先会被放进固定大小的队列里，然后他们延迟字段里 那个指定的值延迟，接下来才送到他们的目的地。
掩码	无 如果选择了"源"或"目的"，将会创建一个将针对每一个遇到的源/目的IP地址的管道，这个管道会与上面给定的带宽、延迟、丢包率和队列大小一起被创建。 这个掩码可能会方便得指定每个主机的带宽限制
描述	m_Total Download 你可以为了参考 输入一个描述（不是必须的）。

保存

5.6.2 添加队列

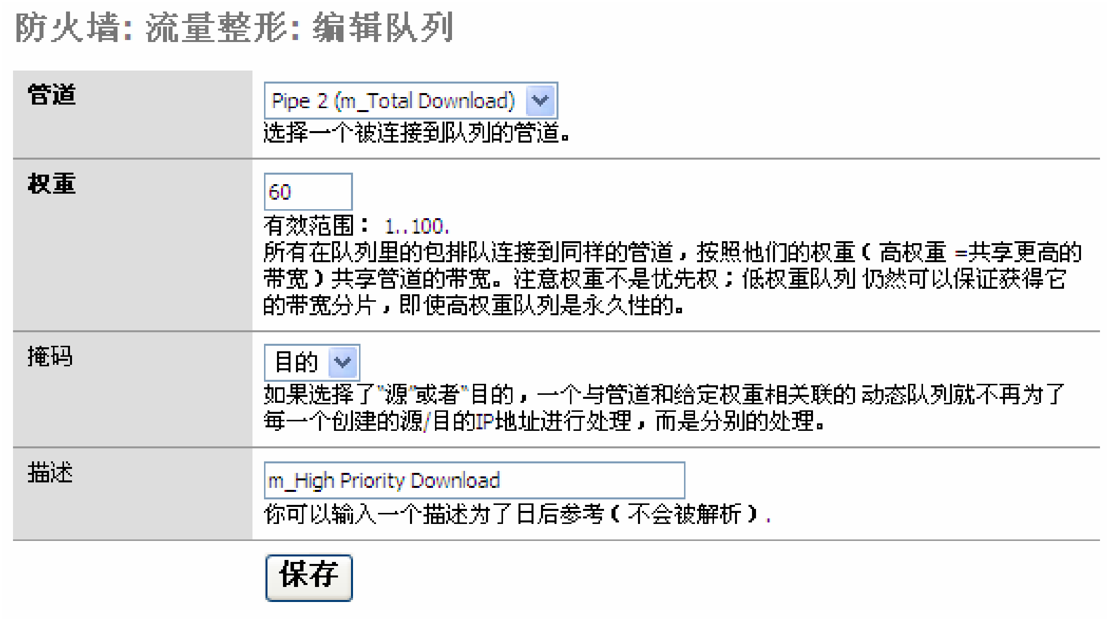
防火墙: 流量整形: 队列

规则 | 管道 | **队列** | 魔术整形向导

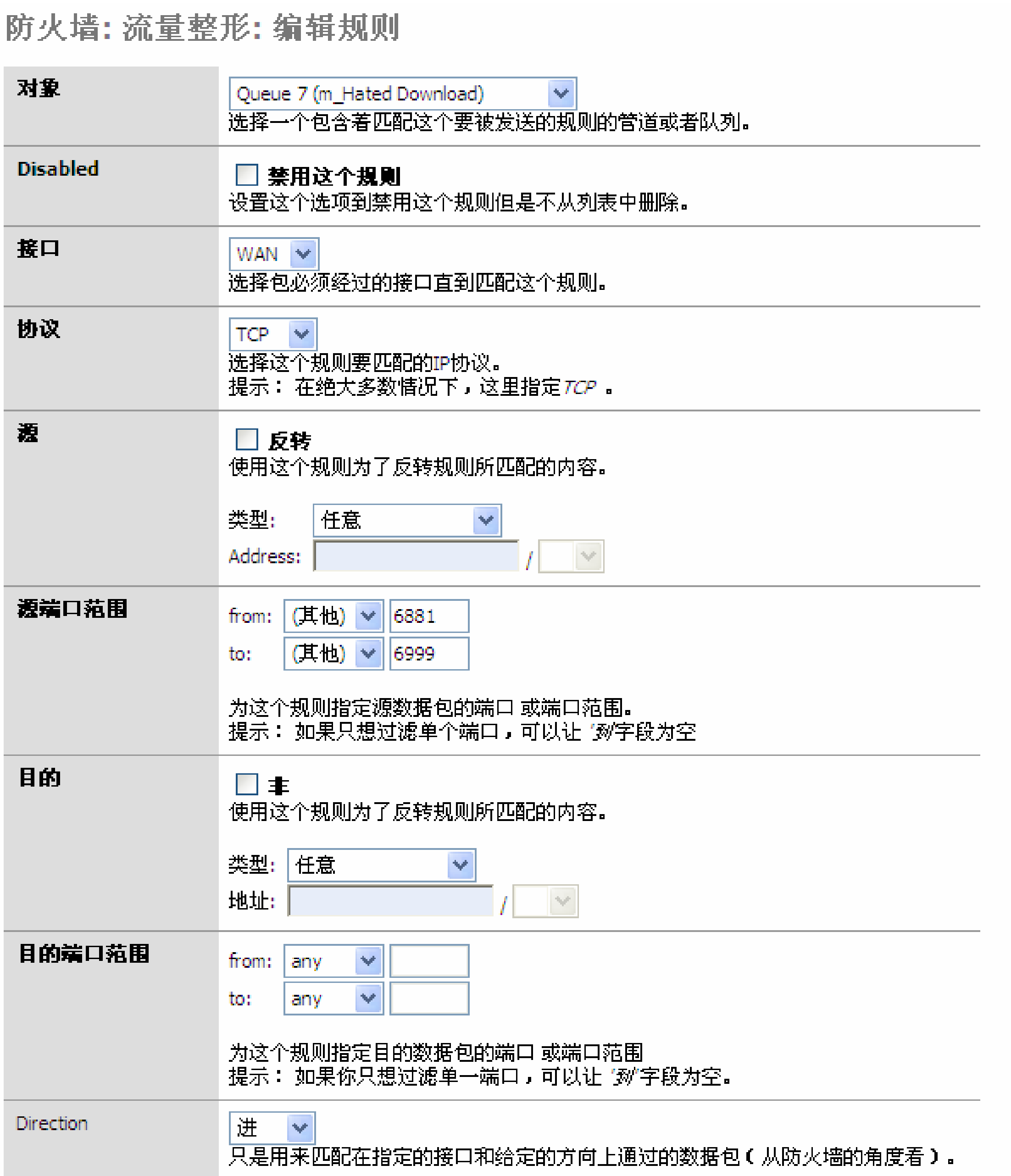
序号	管道	权重	掩码	描述
1	m_Total Upload	50	source	m_High Priority #1 Upload
2	m_Total Upload	30	source	m_High Priority #2 Upload
3	m_Total Upload	15	source	m_High Priority #3 Upload
4	m_Total Upload	4	source	m_Bulk Upload
5	m_Total Upload	1	source	m_Hated Upload
6	m_Total Download	30	destination	m_Bulk Download
7	m_Total Download	10	destination	m_Hated Download
8	m_Total Download	60	destination	m_High Priority Download

注意: 当一个队列 不被任何规则引用的时候才可以删除。

点击“+”按钮，可以添加新的队列：

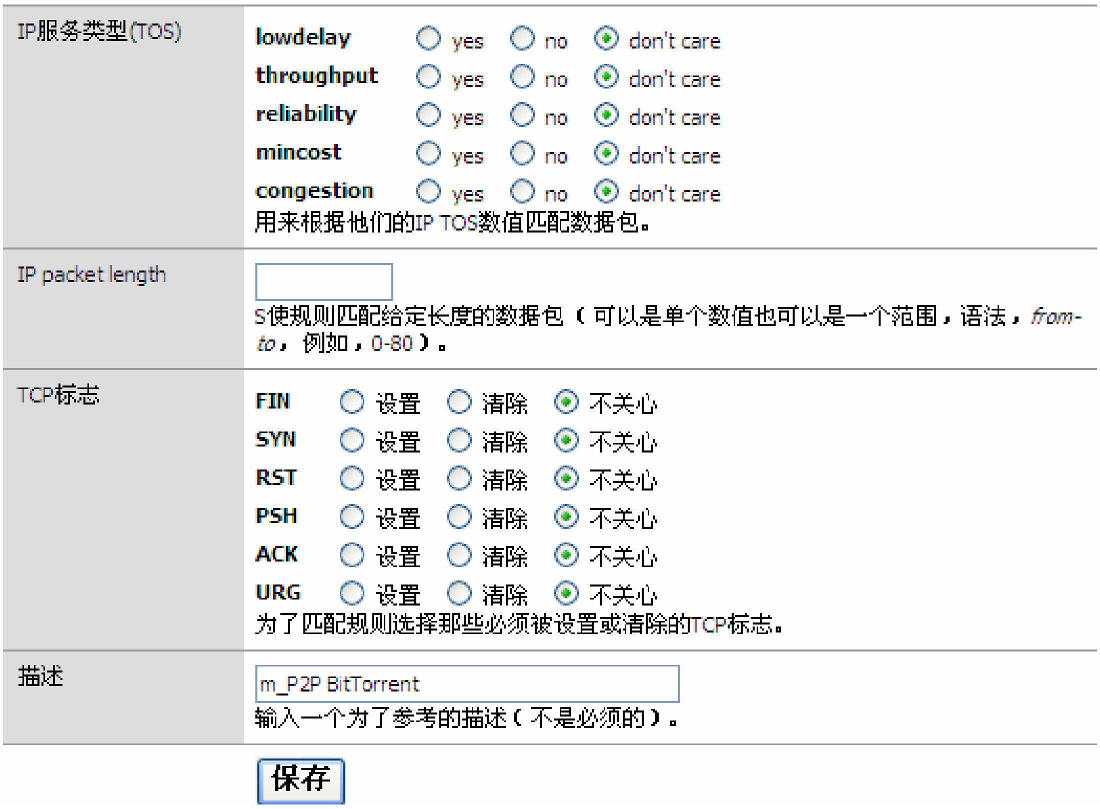


5.6.3 添加规则



防火墙: 流量整形: 编辑规则

对象	Queue 7 (m_Hated Download) 选择一个包含着匹配这个要被发送的规则的管道或者队列。
Disabled	☐ **禁用这个规则** 设置这个选项到禁用这个规则但是不从列表中删除。
接口	WAN 选择包必须经过的接口直到匹配这个规则。
协议	TCP 选择这个规则要匹配的IP协议。 提示： 在绝大多数情况下，这里指定*TCP* 。
源	☐ **反转** 使用这个规则为了反转规则所匹配的内容。 类型: 任意 Address: /
源端口范围	from: (其他) 6881 to: (其他) 6999 为这个规则指定源数据包的端口 或端口范围。 提示： 如果只想过滤单个端口，可以让 '到'字段为空
目的	☐ **非** 使用这个规则为了反转规则所匹配的内容。 类型: 任意 地址: /
目的端口范围	from: any to: any 为这个规则指定目的数据包的端口 或端口范围 提示： 如果你只想过滤单一端口，可以让 '到'字段为空。
Direction	进 只是用来匹配在指定的接口和给定的方向上通过的数据包（从防火墙的角度看）。


IP服务类型(TOS)
lowdelay yes no don't care
throughput yes no don't care
reliability yes no don't care
mincost yes no don't care
congestion yes no don't care
用来根据他们的IP TOS数值匹配数据包。
IP packet length
S使规则匹配给定长度的数据包（可以是单个数值也可以是一个范围，语法，from-to，例如，0-80）。
TCP标志
FIN 设置 清除 不关心
SYN 设置 清除 不关心
RST 设置 清除 不关心
PSH 设置 清除 不关心
ACK 设置 清除 不关心
URG 设置 清除 不关心
为了匹配规则选择那些必须被设置或清除的TCP标志。
描述
m_P2P BitTorrent
输入一个为了参考的描述（不是必须的）。
保存

5.6.4 整形向导



5.7 别名

在 WEB 界面中，所有浅蓝色背景的输入框都可以输入别名。以下是防火墙规则编辑界面，其中的源、目的地址可以选择使用别名：



别名就象一个占位符，使用别名定义各种规则，当别名代表的 IP 或网络改变时，可以避免直接修改规则本身，只需要修改别名的属性即可。所有浅蓝色背景的输入框都可以输入别名，根据别名列表对所输入的别名进行解析，如果解析失败，则认为输入了非法值。

5.7.1 输入别名

点击防火墙->别名页面中的⊕，可以添加新别名：



5.7.1.1 名称

别名的名称，可以在整个系统中所有浅蓝色背景的输入框中使用。

5.7.1.2 类型

可以选择主机或网络。

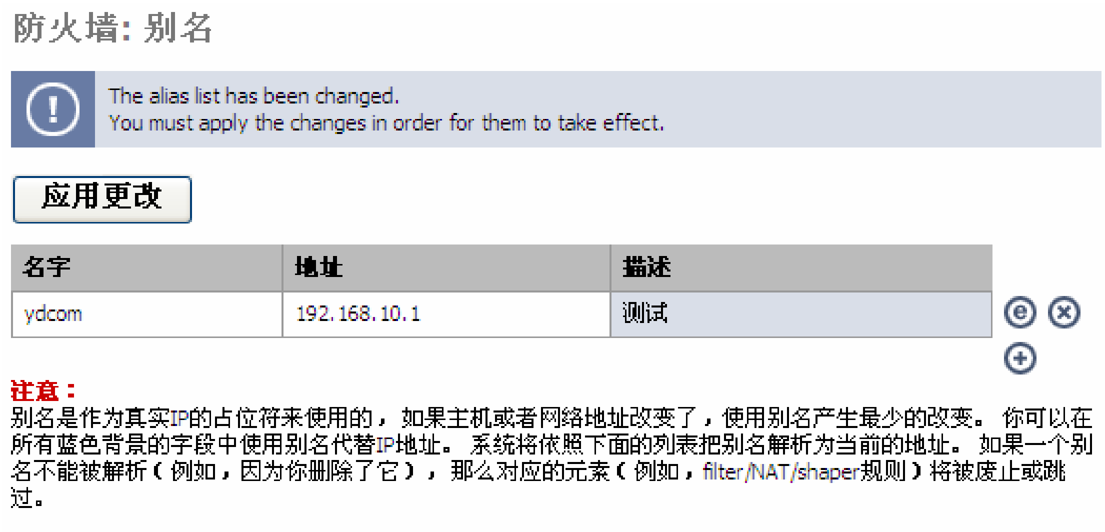
5.7.1.3 地址

输入别名代表的 IP 地址或子网。

5.7.1.4 说明

为了便于管理，建议输入相关的说明。

输入所有参数，点击“保存”，然后“应用”所作的修改：



5.7.2 使用别名

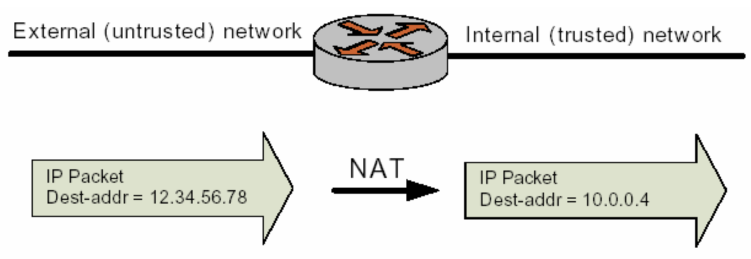
在整个系统中所有浅蓝色背景的输入框中,选择“单个主机或别名”, 然后就可以输入已经定义

的别名。

第六章 关于 NAT(略)

NAT（Network Address Translation）允许内部网络中使用私有 IP（RFC 1918）的机器通过一个公网 IP 访问互联网。

当网络流量通过 NAT 设备时，NAT 服务可以修改网络 IP 包的地址和端口信息。NAT 设备可以是数据包的源或目的，也可以源和目的之间的设备。如图：



NAT 最早设计用来节省 IP，满足越来越多设备访问互联网的需求。但它对于网络安全也是非常重要的。

内部网络中的计算机可以使用任何保留的私有 IP（由 IANA(Internet Assigned Numbers Authority)保留作私用，见 RFC 1918），这些保留 IP 不用在互联网中，所以外部机器不能对其直接路由。下面是保留的私有地址：

■ 10.0.0.0 到 10.255.255.255(CIDR: 10.0.0.0/8)

■ 172.16.0.0 到 172.31.255.255(CIDR: 172.16.0.0/12)

■ 192.168.0.0 到 192.168.255.255(CIDR: 192.168.0.0/16)

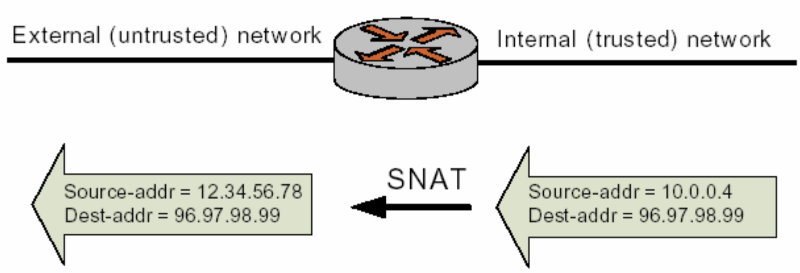
支持 NAT 的路由器用自己的公网 IP 地址替换内部的私有地址，从而把内部机器的 IP 地址对外部网络隐藏起来（相隔离），只有路由器中的公网 IP 才对外部网络开放。使用多线路的情况下，路由器可以管理多个公网 IP，并动态地选择一个进行 NAT 替换。

要明确一点，虽然 NAT 可以很大程度地减少内部机器与外部网络建立不安全的连接，但它不提供内部机器对外连接的保护，这种连接可以出于任何目的。因此 NAT 要与其它数据过滤功（防火墙）能一起使用，才能给网络提供周全的保护。

NAT 的分类

■ 源 NAT（S-NAT,SNAT）

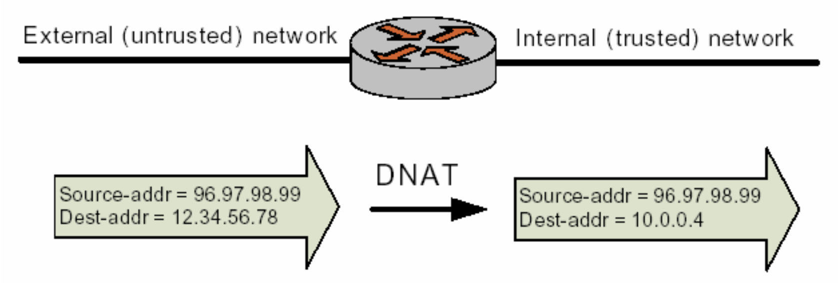
这是 NAT 是最常见形式，用于内部主机需要主动连接外部主机。SNAT 位于内部网络进入外部网络的出口处。先进行路由再进行 SNAT 操作，从内部网络到外部网络数据包的源地址被替换，如图：



还有一种特殊的SNAT形式，称作masquerade，它使用对外接口的主IP替换出站(outgoing)数据包的源IP。

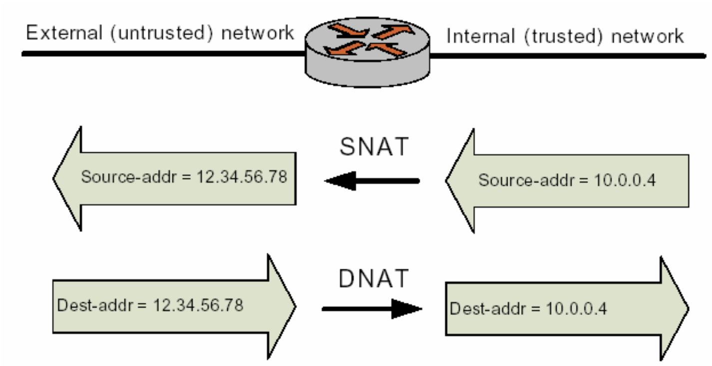
■　目的 NAT（DNAT，端口映射）

DNAT 用于外部主机主动建立与内部主机的连接，如内部网络中的 FTP 服务。SNAT 位于外部网络进入内部网络的入口处。先进行 SNAT 操作再进行路由。从外部网络到内部网络数据包的目的地址被替换，如图：



■　双向 NAT（Bi-Directional NAT）

当同时配置了源 NAT 和目的 NAT，其结果就是双向 NAT。用于内部主机需要主动连接外部主机，同时外部主机也需要主动连接内部主机的情况。如图：



第七章 关于流量整形(略)

第八章 IPsec

IPsec VPN(Virtual Private Network)公众网络中建立虚拟网络，通过加密隧道保护两端或多端之间通讯的私密性。VPN 提供：

- 数据完整性：确保数据在通过两端间的网络时没有被篡改、修改。数据完整性使用 HASH 算法实现。
- 认证：确保所接收的数据是可信的，也就是说数据是来自所期望的来源，而不是来自任何一个伪装的来源。认证也是使用 HASH 算法实现。
- 保密：保护数据避免在传输过程中被非法查看、拷贝。数据保密通过加密算法完成。

IPsec VPN 保护两端之间的通讯、对资源的访问，使用加密、认证和密钥管理协议。通过正确配置 IPsec VPN，可以保障通讯安全，避免信息受到攻击。

Router/VPN 支持 site-to-site 的 IPsec VPN 连接，提供远程(移动)接入（point-to-site）。

site-to-site VPN 连接一般在两个或多个 VPN 网关之间建立，实现两端中的用户主机、服务器，以及其它网络设备互联。连接一般是基于一对网络源 IP、目的 IP 之间建立，允许两端中的多台主机共享同一个隧道。我们还支持使用一对域名建立连接，以适应动态 IP 接入互联网的应用。

site-to-site VPN 可以让企业在不同办公场地间建立低成本的连接，经常用来替换非常昂贵的专用线路或帖中继。

8.1 IPsec 结构简介

IPsec 是一套位于网络第三层的协议，使用加密和认证技术，设计用于保护端对端(end-to-end)通讯的安全。在 IP 网络设备看来，经加密、封装的 IP 包可以象普通的 IP 包一样进行路由，只有 IPsec 端点的设备需要实现 IPsec 协议。

IPsec 结构包含主要的组件，它们是：

- 认证头协议（AH:Authentication Header Protocol）
- 封装安全载荷协议(ESP：Encapsulating Security Payload Protocol)
- Internet 密钥交换协议(IKE，或旧称为 ISAKMP/Oakley)

支持 ESP 和 IKE。ESP 加密载荷数据包，避免受到监视，而 IKE 提供了安全交换加密密钥的方法，以及提供了认证、加密协商的方法。

IPsec 端点间需要协商一组安全参数，两端的参数必须匹配。然后它们建立安全联盟（SA：Security Association）。一个 IPsec SA 只描述 IPsec 连接的一个通讯方向。在 IPsec 连接的两个方向传输数据包，要求有 入站 SA（inbound SA）和出站 SA(outbound SA)。

8.2 IPsec 阶段 1 和阶段 2

建立 IPsec 连接需要两个阶段，称为 IKE 阶段：

- IKE 阶段 1. 两端互相认证，协商密钥（DH）。其结果是建立一个加密的隧道供阶段 2 协商 ESP SA。
- IKE 阶段 2. 两端使用阶段 1 建立的加密隧道协商 ESP SA。真正用来加密两端之间通讯数据的是 ESP SA。

IKE 阶段 1 建立一条 ISAKMP SA(现在一般称为 IKE SA)。IKE 协议用来动态协商、认证密钥信息，以及其它保障通讯安装的参数。(IKE 本身使用四个协议的组合（包括 ISAKMP 和 Oakey）在 IPsec 的上下文中动态管理密钥。)

如果 IKE 阶段 1 协商成功，就可以建立 ISAKMP SA（IKE SA）。ISAKMP SA 实质上包含了从协商（winning proposal）获得的信息，记录了成功协商得到的加密方法以及密钥。这实际上建立了一条安全控制隧道（control channel），其中维护着用来保护阶段 2 协商的密钥和其它安全相关的信息。ISAKMP SA 只加密 ESP SA（Security Association）的协商过程,以及两端之间的 IKE 消息包。

一条 ISAKMP SA 有一个预先指定的生存期（lifetime）。这个生存期只是在本端配置，不是通过协商或两端之间传递来确定的。两端所配置的生成期可以不相同。当配置的生存期到期，将会协商一条新的 ISAKMP SA。

IKE 阶段 2 协商也上由 IKE 协议管理。使用（IKE）SA 提供的加密方法，根据安全策略尝试协商一条阶段 2 SA。安全策略的信息包括通讯主机及其子网，以及为这个连接提供安全服务的 ESP 信息，如加密算法和 HASH 算法。如果 IKE 阶段 2 协商成功，将在两个端点间建立一对 ESP SAs(现在一般称 IPsec SAs)，一条入站（inbound），一条出站（outbound）。这就是两端点间的加密 VPN 通道。从这一刻开始，用户数据可以经加密隧道中交换。

在任意两个 IPsec VPN 端点中，可以只有一条“控制隧道（control channel）”用来交换阶段 2 的密钥。这意味着任意两个端点之间，只有一条 ISAKMP SA。

在两个 VPN 端点间，可以定义任意数量的安全策略。例如，可以定义一个建立两主机间隧道的安全策略，也可以定义一个建立主机和子网间隧道的安全策略，或两个子网间的。由于两端间可以存在多个隧道，意味着两端间可以有多条 IPsec SA 被建立、激活。

8.3 IKE KEY 的交换

为了能够建立一条 ISAKMP SA，两个设备必须达成以下的协议：

- 加密算法
- 加密位数（Diffie-hellman Group）
- 认证方法

- HASH 算法
- 认证的信息（如预共享密钥）

所有这些信息都在IKE阶段1获得，VPN网关可配置多个阶段1建议(proposals)。注意到SA的生存期不经过协商。

在IKE的KEY交换期间，其中一个设备（initiator）首先发出一个请求包，该数据包由该VPN端点中所有阶段1的建议（proposals）按顺序组成。这个建议的集合通知 VPN 的另一端关于本端所支持的安全和认证策略。 另一个设备（Responder）检查这个建议集合，并返回两个设备所能达成的安全性最高的策略。如果这个处理成功 ,两个设备对安全参数达成一致，ISAKMP就建立了。

一旦ISAKMP SA建立起来，两个设备可以用这个IKE SA 加密阶段2的流量。阶段2期间，两端尝试为各自配置的每一条匹配的安全策略协商一个IPsec SA。仅当两端都建立了自己的IPsec SA， 才能传输IPsec流量。

不同的设备发起IKE协商的时机也是不同的。许多VPN设备是按需（on-demand）建立VPN隧道的。这些设备监视IP流量，检查是否“感兴趣”的，如果是，则再看一下是否与一条预先配置的安全策略匹配。一旦该设备收到满足某一条安全策略的IP流量，它就会尝试协商一条IPsec SA用来加密那些IP流量。

其它设备，包括我们的系统，只要输入一条正确的安全策略，就会开始IKE两阶段协商。如果两端都是这样做，会出现竞赛的情况，导致建立重复的IPsec SA（但这并不影响两端的

通讯）。

8．4 加密算法

加密算法用来加密数据，让它在传输过程不能被监视或查看。支持以下的加密算法：

- DES
- AES
- 3DES
- SSF33/SCB2: 国家加密标准

8．5 HASH 算法

HASH函数是一种密码算法，用于消息包的认证。HASH函数可以根据输入的任意长度的消息包产生固定长度的输出，称为消息摘要或指纹（message digest/fingerprint）。HASH函数用来校验消息包是否经过篡改（未经允许的）。

支持以下的HASH函数：

- MD5: 128bits(16bytes)
- SHA1: 160bits(20bytes)

8．6 预共享密钥

预共享密钥（PSK: Pre-Shared Key/Pre-shared secret）是一种认证方法。密钥本身是一个字符串，由通讯双方预先达成协议使用该密钥来认证会话。它被用来产生一个HASH，VPN两端据此HASH进行相互认证。

注意到密钥本身虽然是一个字符串，但它不是一个“口令”。实际上它被用来产生一个HASH KEY，一个用来校对两端的指纹。这意味着较长、较复杂的字符串会比短字符串安全。应该选择复杂的PSK，避免使用短的，因为它们更容易受到攻击。

PSK在IKE协商过程中不会被传输，它在两端配置且必须相同（匹配）。

PSK 是对称加密的一个例子：KEY在通讯双方都一样。对称加密与非对称加密相比，计算量要少得多，因此也更加快。然而，对称加密算法，要求通讯双方交换KEY（密钥），如何安全交换KEY是一个问题。

PSK 和数字签名是最常用的IKE认证方式。PSK是一个简易、高效方法，可以很方便地设置认证。然而它有以下缺点：

- 如果PSK泄漏出去而没有人发现，那么攻击都就可以一直访问你的网络，只要该PSK还在使用。
- PSK都是经手工配置，而且应该经常改变。然而，网管经常会由于太忙而把这事忘记。与远程用户一起使用PSK，无异于给他们一个进入你网络的口令。

建议在可能的情况下，不使用PSK。

8.7 数字签名

与PSK一起，RSA数字签名也是最常用的IKE认证方式。

RSA 数字签名的加密密钥由两部分组成：公钥部分和私钥部分。公钥部分是对外公开的，甚至可以对公众分发。私钥部分必须保密。这些密钥（KEY）虽然有数学的联系，但它们是独立的，不能进行互相推导。

RSA 的密钥（KEY）可以用来加密或认证，这基于基于两个事实：

- 用公钥加密的数据只能使用对应的私钥解密。任何一端可以安全地发送经过公钥加密的数据，然后把它发送到私钥的持有人。
- 经过HASH处理的数据可以使用私钥加密，经过这样处理的数据就叫经过数字签名了。因为任何人都可以使用公钥进行校验该数字签名。

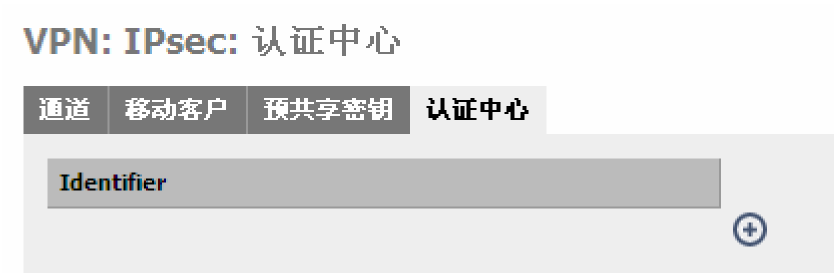
使用RSA密钥进行加密非常安全，但也极慢，以至于使用它来加密所有数据不现实。实际应用中，采用以下的替代方案：

- 使用HASH函数处理数据，产生消息摘要或指纹（message digest/fingerprint）。指纹的长度比原来的数据一般要短很多（MD5只有16字节，SHA1只有20字节）。任何其他人使用同样的HASH函数可以产生相同的指纹。
- 使用私人密钥加密生成的指纹。这个经加密的指纹就是数字签名。
- 原来的数据和产生的数签名（经加密后）一起发送到对端（其他人）。
- 对端收到数据包，对其解密。然后使用（与私钥对应的）公共密钥解密数字签名，重新得到指纹。
- 对端使用相同的HASH函数处理接收到的数据，同样得到一个指纹。比较恢复出来的指纹与新生成的指纹：
    - 如果两个指纹匹配，则说明数据确实来自私钥的持有人，可以接受该数据包。

■ 如果两个指纹不匹配，说明数据可能被篡改，拒绝该数据包。

8.8 证书管理

认证中心：集中管理可信任对象的公共密钥，采用X.509 PEM 格式的证书。这些证书将在IKE阶段1期间用作身份认证。



点击⊕可以添加新的数字证书:



8.9 Diffie-Hellman Groups

DH密钥交换协议可以安全地在非安全的公众网络（如互联网）交换加密密钥。DH密钥交换协议由WhiteField Diffie和Martin Hellman于1976年发明，它基于以下两个事实：

■ 非对称加密算法比对称加密算法更安全，后者需要通讯双方交换加密密钥。

■ 然而，非对称算法要比对称算法慢得多，前者需要大量的计算。

DH密钥交换的原理：

- 通讯双方A,B商定两个大素数（DH Group）n,g(n = gk+1）;
- A 随机选取一个大数x, 计算X = g^x mod n, 把 X 发送给对方B ;
- B随机选取一个大数y, 计算Y = g^y mod n, 把 Y 发送给对方A ;
- A计算密钥：k1 = Y^x = (g^y mod n)^x = g^xy mod n;
- B计算密钥：k2 = X^y = (g^x mod n)^y = g^xy mod n;

其结果是，通讯双方A、B都独立地计算出相同的密钥。其中x,y可以直接使用RSA中的私钥。

DH密钥交换的安全性可以这样理解，n,g 是公开的，网络上可以监听到 n,g,X,Y，但是从这个数导出x,y非常难（不可能），从而使用 k1/k2（两个值相同，独立计算）进行加密是安全的。

在IPsec应用中，DH 应用在IKE阶段1，实现在互联网中安全地交换对称加密密钥，建立IKE SA。一旦完成对称加密密钥的交换，在IKE 阶段2 就可以使用对称加密，建立IPsec SA。

Router/VPN支持以下DH Groups:

- Group 2
- Group 5

8.9 IKE 主模式(Main Mode)

一般情况下，建立ISAKMP SA需要交换6个数据包：

- 头两个数据包决定通讯策略；
- 接着两个数据包交换DH公共数据
- 最后两个数据包认证DH密钥交换

这是建立IKE阶段1加密通道的正常过程，称为主模式（main mode）。它保证了最高的安全和私密性，因为直到完成DH密钥交换才交换认证信息，那时已经可以加密了；主模式也提供了最大的灵活性，因为可以交换更多的信息来协商合适的选项（策略）。另一方面，主模式比较慢，因为在交换信息过程中需要进行多次计算量很大的模幂运算。

8.10 IKE 主动模式（Aggressive Mode）

IKE采用主模式协商，将产生一定的延时，有些厂商设备希望采用交换更少的数据包、更快的模式，这种模式称为主动模式（Aggressive Mode）。这种模式下只需要交换3 个数据包：

- 头两个消息决定通讯策略，同时交换DH公众数据。所有与安全联盟（SA）、密钥交换以及认证相关的信息都一次性传输。
- 第三个数据包用于认证响应者（Responder），同时也结束协商。

8.11 IKE 对话过程

参考 RFC2409。

IKE对话过程用到的符号约定：

- HDR: 表示ISAKMP的包头, HDR* 表示包的内容被加密
- SA: 表示用于协商的数据，其中包含一个多个协商建议；响应者（Responder）必须只回应一个方案。
- <P>_b: 表示载荷<P>的数据部分
- SAi_b: 表示SA载荷完整的数据。如：IPsec 的DOI(Domain of Information)、situation、建议及所有发起者提供的加密算法。
- CKY-I,CKY-R: 分别表示发起者(Initiator)与响应者(Responder)的cookie，不包括ISAKMP包头。
- g^xi,g^xr: 分别表示发起者(Initiator)与响应者(Responder)的Diffie-Hellman的公开值（xi,xr是随机数）
- g^xy：表示经Diffie-Hellman交换后得到的密钥。
- KE: 包含Diffie-Hellman公共数的载荷。
- Nx: 表示当前载荷，其中x可以是: i,r，分别表示发起者(Initiator)与响应者(Responder)
- IDx: 表示鉴定载荷，其中x 可以是: ii,ir,分别表示发起者(Initiator)与响应者(Responder)阶段1的协商；或者，ui,ur, 分别表示发起者(Initiator)与响应者(Responder)阶段2的协商.
- SIG: 表示签名载荷
- CERT： 表证书载荷
- HASH: 表HASH载荷
- prf(key,msg): 表示随机数函数
- SKEYID：表示一个源自密钥的字符串，只有参与密钥交换双方才知道的。
- SKEYID_e：表示被ISAKMP SA用来加密的相关数据

- SKEYID_a: 表示被ISAKMP SA用来认证的相关数据
- SKEYID_d: 表示用来派生密钥的相关数据
- <x>y: 表示用密钥“y”来加密“x”
- | ：表示两个条件的合并，如X|Y

IKE中，SKEYID的计算方法由授权方法决定，使用上面的符号可以表示如下：

数字签名:　　　　　SKEYID = prf(Ni_b | Nr_b, g^xy)

RSA公钥加密:　　　　SKEYID = prf(hash(Ni_b | Nr_b), CKY-I | CKY-R)

预共享密钥:　　　　SKEYID = prf(pre-shared-key, Ni_b | Nr_b)

IKE协商(主模式或主动模式)最终得到一组加密相关的数据，使用上面的符号表示如下：

SKEYID_d = prf(SKEYID, g^xy | CKY-I | CKY-R | 0)

SKEYID_a = prf(SKEYID, SKEYID_d | g^xy | CKY-I | CKY-R | 1)

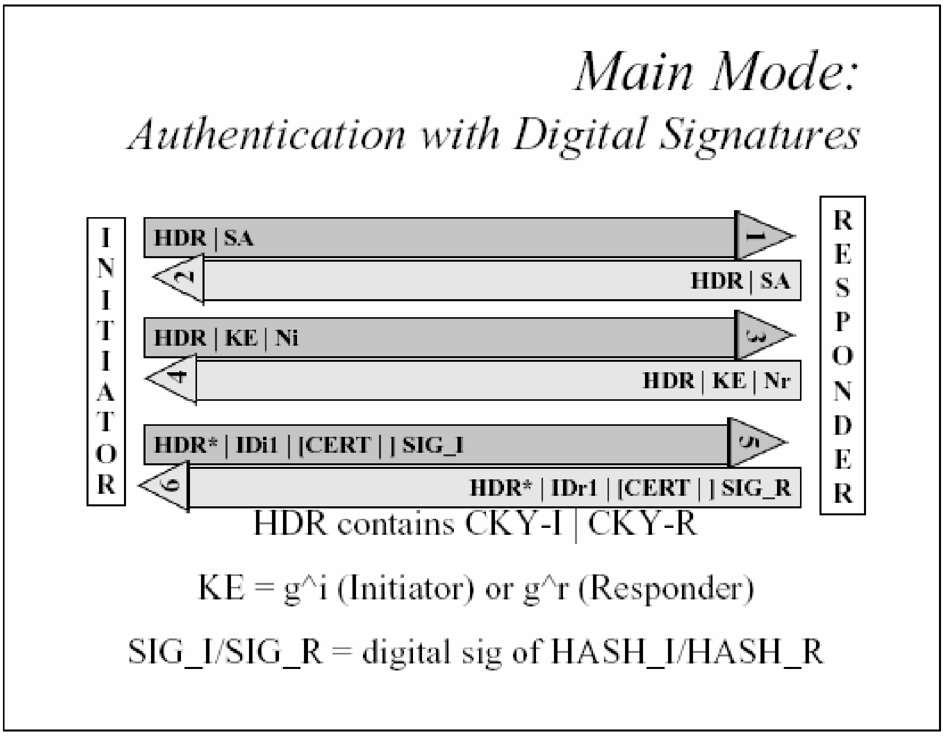
SKEYID_e = prf(SKEYID, SKEYID_a | g^xy | CKY-I | CKY-R | 2)

用作认证的HASH产生方法，表示为：

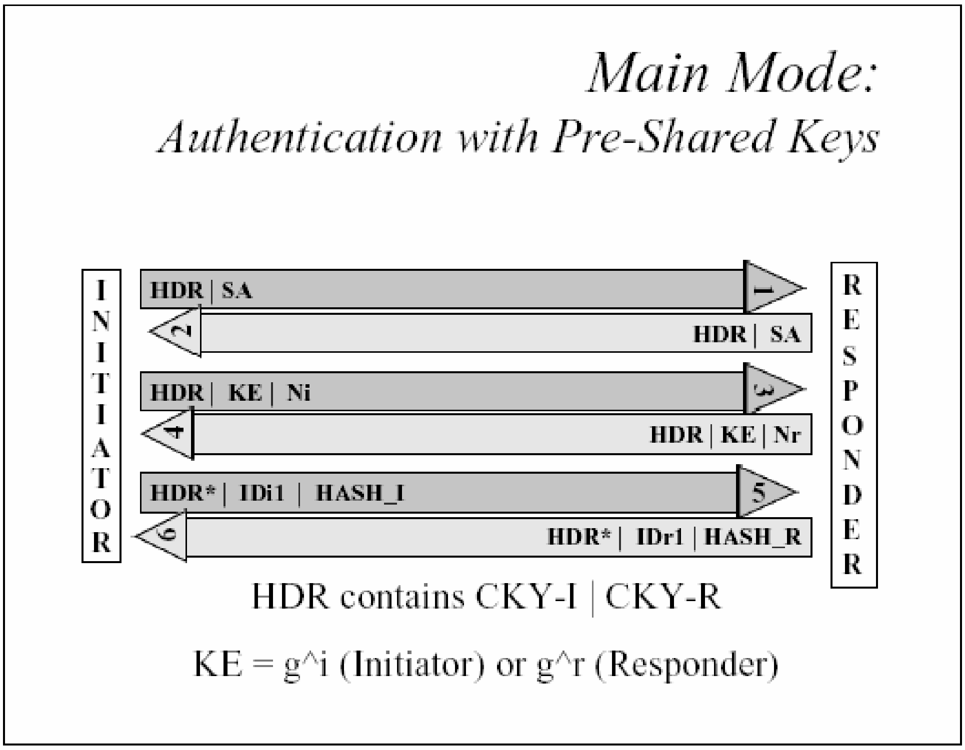
HASH_I = prf(SKEYID, g^xi | g^xr | CKY-I | CKY-R | SAi_b | IDii_b)

HASH_R = prf(SKEYID, g^xr | g^xi | CKY-R | CKY-I | SAi_b | IDir_b)

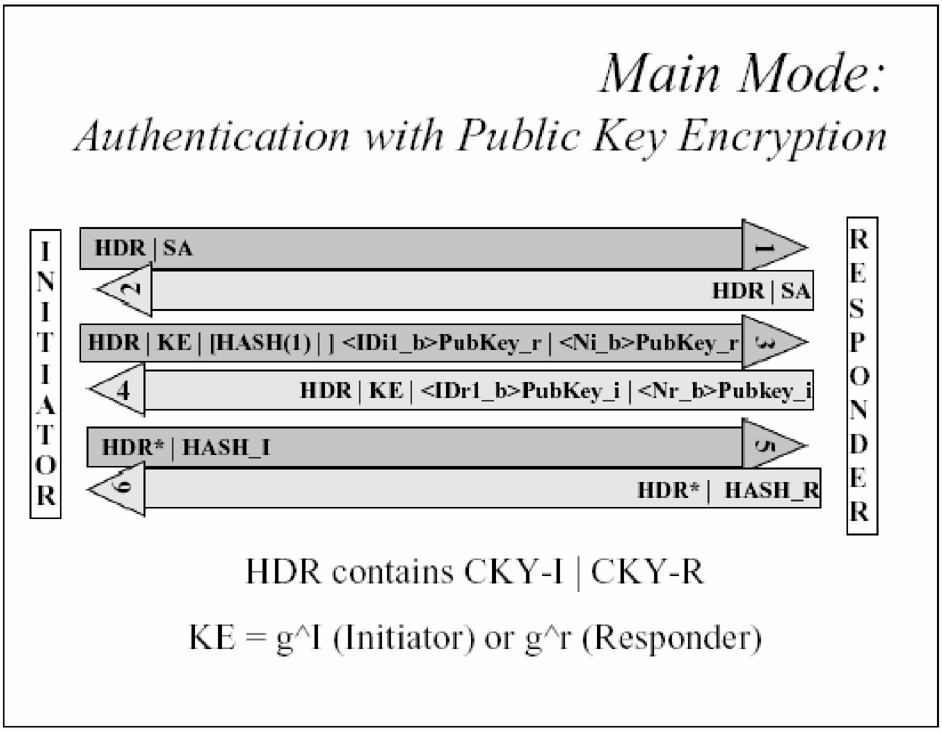
阶段 1：主模式+数字签名



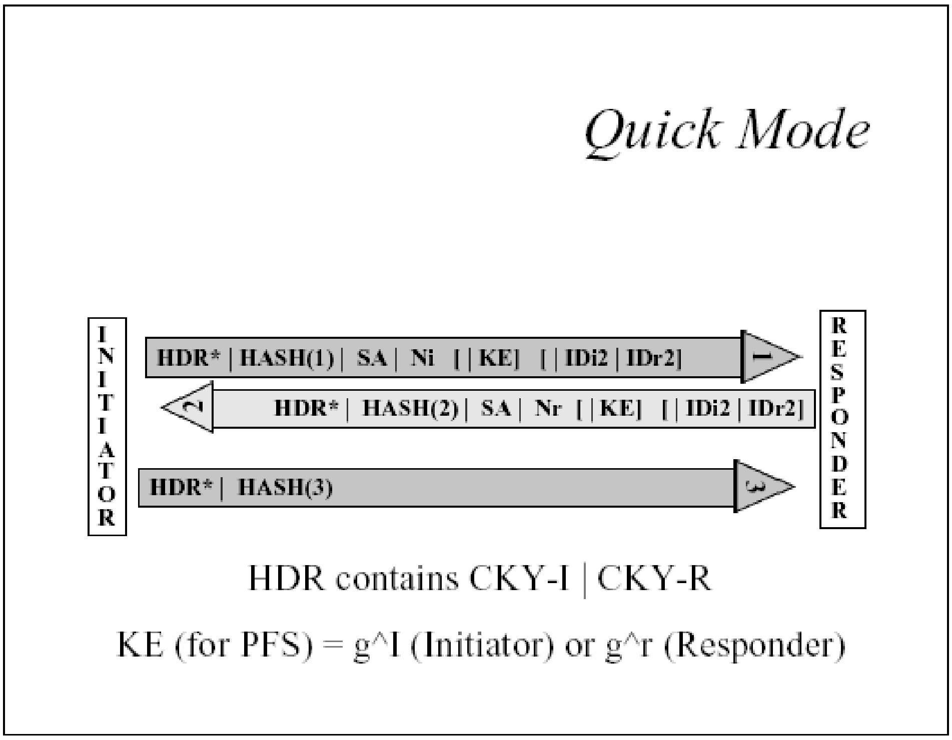
阶段 1：主模式+预共享密钥



阶段 1：主模式+公共密钥



阶段 2：快速模式(Qick Mode)



其中 KE 是可选的，当启用 PFS 时:

- 如果包含 KE，则：KEYMAT = prf(SKEYID_d, g(qm)^xy | protocol | SPI | Ni_b | Nr_b)
- 如果不包含，则: KEYMAT = prf(SKEYID_d, protocol | SPI | Ni_b | Nr_b).

8.12 完美向前加密（PFS： Perfect Forward Secrecy）

PFS是指周期性地使用私钥（PSK或RSA）生成临时的加密密钥（会话密钥），该临时密钥使用很短一段间后就被丢弃。后续产生的KEY与之前生成的KEY之间相互独立。使用这种方法，即使某一个加密密钥被破解，也不会影响到使用后续密钥加密的数据安全。

PFS在Route/VPN中作为一人选项。

8.13 IPsec over NAT-T

由于 IPsec 的 ESP 载荷加密封装了 TCP/UDP 数据包，穿越 NAT 时无法根据 UDP/TCP 端口进行地址转换，导至 IPsec 通讯失败。

目前较通用的解决方案是使用 NAT-T 协议(方法)。该方法修改了 IKE 阶段 1、阶段 2(Quick Mode)的通讯协议，达到检测 NAT、协商使用 NAT-T 方法的目的。NAT-T 的规范在以下两个 INTERNET-DRAFT 定义：

draft-ietf-ipsec-nat-t-ike-06.txt

draft-ietf-ipsec-udp-encaps-06.txt

目前，FreeBSD6 已经有 IPsec over NAT-T 的补丁:

racoon-nattraversal-freebsd6.patch

这里我们直接集成该补丁。

8.13.1 NAT-T 协议包

这里给出 NAT-T 中定义的数据包，具体说明参考上述两个 INTERNET-DRAFT。

- NAT-D payload(NAT Discovery PayLoad)

```
  1 2 3 4 5 6 7 8 1 2 3 4 5 6 7 8 1 2 3 4 5 6 7 8 1 2 3 4 5 6 7 8
 +---------------+---------------+---------------+---------------+
 | Next Payload   |      RESERVED      |            Payload length              |
 +---------------+---------------+---------------+---------------+
 ~                          HASH of the address and port                          ~
 +---------------+---------------+---------------+---------------+
```

- UDP-encapsulated ESP Header Format

```
  0                   1                   2                   3
  0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|        Source Port            |      Destination Port         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|           Length              |           Checksum            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                  ESP header [RFC 2406]                        |
~                                                               ~
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

- Floated IKE Header Format

```
  0                   1                   2                   3
  0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|        Source Port            |      Destination Port         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|           Length              |           Checksum            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                      Non-ESP Marker                           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

```
|                  IKE header [RFC 2409]                        |
~                                                               ~
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

- NAT-keepalive Packet Format

```
  0                   1                   2                   3
  0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|        Source Port            |      Destination Port         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|           Length              |           Checksum            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    0xFF       |
+-+-+-+-+-+-+-+-+
```

- Transport Mode ESP Encapsulation

```
          BEFORE APPLYING ESP/UDP

      ----------------------------

IPv4  |orig IP hdr  |     |      |

      |(any options)| TCP | Data |

      ----------------------------

          AFTER APPLYING ESP/UDP

      -------------------------------------------------------

IPv4  |orig IP hdr  | UDP | ESP |     |      |   ESP   | ESP|

      |(any options)| Hdr | Hdr | TCP | Data | Trailer |Auth|

      -------------------------------------------------------

                                |<----- encrypted ---->|

                          |<------ authenticated ----->|
```

- **Tunnel Mode ESP Encapsulation**

```
          BEFORE APPLYING ESP/UDP

      ----------------------------

IPv4  |orig IP hdr  |     |      |

      |(any options)| TCP | Data |

      ----------------------------
```

```
                 AFTER APPLYING ESP/UDP

      -------------------------------------------------------------

IPv4 |new h.| UDP | ESP |orig IP hdr   |        |          |     ESP     | ESP|

      |(opts)| Hdr | Hdr |(any options)| TCP | Data | Trailer |Auth|

      -------------------------------------------------------------

                             |<------------ encrypted ----------->|

                       |<------------- authenticated ------------>|
```

8.14 IPsec 使用动态域名

IPsec 的语义，其隧道的定义要求对端必须使用 IP 地址。在实际应用中，服务商一般为用户分配动态 IP，如 ADSL 拨号。

为了让 IPsec 能在动态 IP 的环境中应用，需要提供一种机制，在建立 IPsec 隧道时，能找到对端当前的 IP。目前常用的方法使用动态域名。

系统实现一个 ddnsguard 进程，它维护所有 IPsec 相关的域名（远端）, 定时检查相关域名是否发生变化。一旦远端 IP 发生变化，ddnsguard 进程重建受影响的 IPsec 隧道。

在 IPsec 定义中，其中**远程网关**可以使用动态域名，如图：



8.15 配置 IPsec VPN 隧道

登录，选择 VPN 下的 IPsec，点击⊕，可以添加 IPsec VPN 隧道：



编辑隧道页面显示一个很大的配置页面，这里我们分几部分别说明。

第一部分：定义 IPsec 隧道的网络范围



上面是我们需要关注的第一部分参数，如果这里的参数不正确，将影响阶段 2 连接的建立。这里将指出需要特别注意的参数。

- **模式**：固定设置为“隧道”，不需要修改（界面也不允许修改）。
- **使无效**：这是隧道的一个开关按钮，可以根据需要禁用这个隧道。可以在 VPN:IPsec 页面点击进入编辑页面，“打钩”这个开关，然后“保存”->“应用修改”，就可以暂时关闭一个 IPsec 隧道。如果需要重新启用，采取相反的操作。
- **接口**：选择 VPN 隧道的网络端点，可以选择 WAN，LAN 或 OPT 接口。如果与远端服务器连接，选择 WAN。
- **NAT-T**： 启用 NAT-T 功能，这时 IKE 将自动检测我们的设备是否位于 NAT 后，如果是，则使用 UDP 封装 ESP 封包。当设备位于 NAT 后时，需要启动 NAT-T 功能。

当 NAT-T 起作用时，IKE 使用 **UDP 500 和 UDP4500** 进行 IKE 协商，要求防火墙开放 UDP 4500/500 端口。

- **本地子网**：这里设定对端网络经 VPN 隧道能够访问的本地主机、子网，或整个 LAN 网络，可以简单地设置为本地 LAN 的子网，这意味着你的整个 LAN 都可以被对端网络访问。

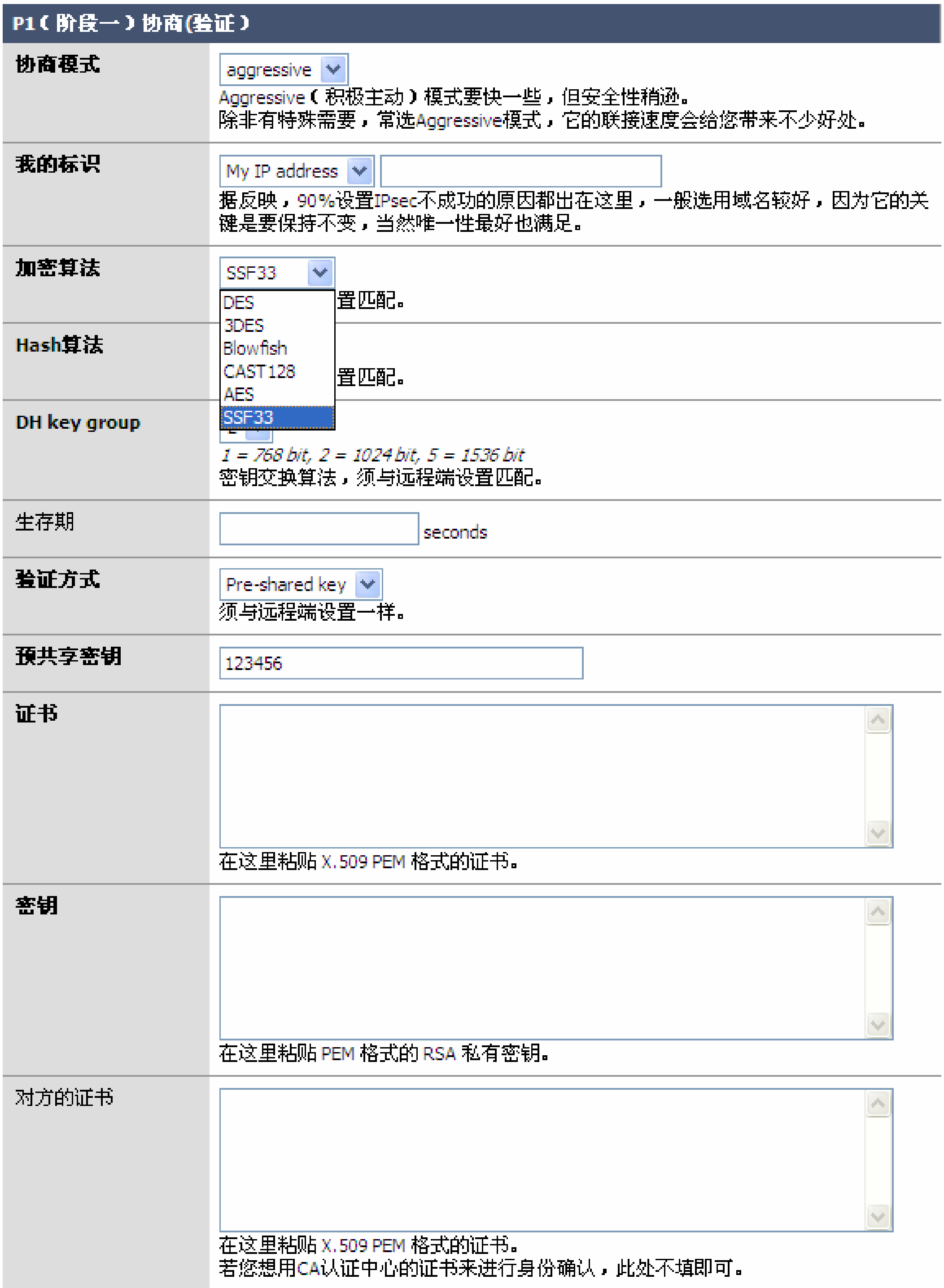
  重要：对端的隧道定义一般也有这一项，必须确保两端的设置一样。例如，如果这里选择"单主机",并输入了一个 IP 地址，对端应该把这个 IP 填入它的"远程子网"中（不同厂家的设备，其名称可能有差异）。对本端，情况也是一样的，这就是下面提到字段。

- **远程子网**：指出本地经 VPN 隧道能够访问的对端主机、子网。同时这项必须与对端的本地子网相同，否则不能建立阶段 2 连接。

- **远程网关**：VPN 隧道的终点，一般情况下是固定（公网）IP。我们的设备可以输入域名，该域可以使用动态域名服务。如果输入域名，必须保证该域名可以被解析。

- **说明**：实际应用中给隧道一些注释是很好的。建议输入关于 VPN 隧道用途、远端信息的说明。

以上是有关建立路由的基本信息，下面说明阶段 1 的授权处理。



配置阶段 1（甚至阶段 2）的参数，最简单的方法是把两端设备的参数配置成完成一样。

- 协商模式：选择 IKE 的安全认证方式。除非你的网络受到贴身的监视，否则应该使用 aggressive（主动模式），由它比主模式(main)快得多，可以保证 VPN 隧道可以快速地重建，从而避免应用超时（需要重建隧道的情况如： 到达 SA 的生命周期）。

- 我的标识：可以选择以下种类：
  - My IP address： 接口的 IP
  - IP Address：指定一个 IP
  - Domain Nmae： 域名
  - User FQDN：用户标识，如 test@yd.com

  建议使用域名标识，这样可以避免由于 IP 改变（如通过 DHCP 获取 IP）而出现问题。
- 加密算法 :3DES 是业界的事实标准 ,这里还支持国家标准算法 SSF33/SCB2。两端 VPN 设备必须使用相同的加密算法。
- HASH 算法：一般使用 MD5，SHA1 相对更可靠，但不是所有设备都支持。两端 VPN 设备必须使用相同的 HASH 算法。
- DH key group:大多设备都至少支持 1024bits（GROUP 2）。使用更多的 bit，安全性也没有提高很多，但却需要更多资源。
- 生存期：与阶段 2 的生存期不一样，这里是指本端等待阶段 1 完成的时间，建议取值 28800。
- 验证方式：可以选择预共享密钥，RSA 签名
- 预共享密钥：两端 VPN 设备必须设置相同的密钥。它大小写敏感，支持特殊字符，建议两者混合使用。
- RSA 签名: 这里可以把 X509 PEM 格式的 RSA 证书粘贴到“证书”、“密钥”输入框。如果选择使用 RSA 签名认证方式，而这里留空，则系统自动使用“认证中心”中的证书。

接下来是阶段 2 的参数设置：



阶段 2 建立实际数据通讯隧道，需要设置协议、隧道的生命周期（没有流量时）:

- 协议： ESP 是 IPsec VPN 传输协议的事实标准，建议使用。提示：正常情况下，系统会自动创建防火墙规则，以便让 ESP，AH 通过。如果系统没有自动建立规则，可能需要手工建立让 ESP/AH 通行的规则。
- 加密算法：可以同时选择多个加密算法，两端 VPN 设备的设置建议保持一致。为简单起见，建议只选择希望使用的一个算法。除了标准算法外，我们的设备还支持国标算法 SSF33/SCB2。
- HASH 算法：可以选择 SHA1，MD5。同样建议只选一个需要的算法，可以的话使用 SHA1 要好一些，两端 VPN 设备的设置必须保持一致。
- PFS key group: 与阶段 1 的含义相同，建议使用 1024bits，缺省关闭。
- 生存期：这是指协商出来的密钥的有效期。不要把这个数值设置得太大，如超过一天（86400），这意味着让其他人有更多的时间破解密钥。也不要设置得小，如 20 分钟，是没有必要的。一般设置为一天就可以。
- 点击保存

- 点击应用修改

第九章 PPTP

这一章基于 Francisoc Artes 的 m0n0wall-PPTP 文档编写，已经被授权。

9.1 前言

这一章的目的是说明几种不同 PPTP VPN 的设置方法，包括 windows XP PPTP 客户端连接到 m0n0wall PPTP VPN 服务的方法，这个文档的后续版本将包括 Linux 和其它平台的客户端。

文档中出现的商标，并不意味着本文档，m0n0wall 或作者与拥有这些商标的公司有任何关系。所有商标的版权属于相关公司所有。

在这一章中，"防火墙"与"m0n0wall"一词的含义相同。主要是因为写"防火墙"会更方便一些。

9.2 读者

需要一点 TCP/IP 的基本认识，了解子网的概念。作者不花力气描述所讨论的项目，但请理解，我只能做这么多。(同时我包含的图片，这样可以省略大量的说明，但使得文档很大)

关于这个文档，如果你有注释，问题或建议，欢迎email到<falcor@netassassin.com>。我会尽快回复，但请发email前通读这一个文档。对于一些常见问题，你也可以在m0n0wall的网站查阅eamail档案。

9.3 前提条件

这个文档中，我们需要几个前提条件。如果你的环境不满足这些条件，需要想办法满足才能让 PPTP 正常工作。

- 你的防火墙的 NAT 功能已经设备，并且可以正常工作，或至少可以按你的路由要求工作。
- 你的防火墙已经配置一个正常工作的网络接口。
- 你有一台客户端机器运行 PPTP 客户端。

下面我们开始说明防火墙的设置。

9.4 子网和 VLAN 路由

这里说的 VLAN 不是指一般意义上的 VLAN，只是由于我们这里需要配置一个虚拟网络，而这个虚拟网络仅当 PPTP 连接建立后才存在。如果有其它更合适的表达方式，我们会修改的。我们需要处理一些虚拟子网，如"远程地址范围"指定子网/28，而 PPTP 客户端收到的掩码却是 2.55.255.255.255，我们只需要忽略这个掩码，而相信 PPTP 隧道的做了正确的事。

"服务器地址"和"远程地址范围"可以使用与 LAN 的相同配置（如在 LAN 配置页面中的 IP 地址和子网位数），我们的例子使用这样的配置。这样做的好处是防火墙将允许源自 VLAN 的流量路由到 WAN 接口（大部分情况是连接到 internet），这样又好又容易；同时也便于人们路由到 WAN（方便 PPTP 接入的用户上网，如果不希望这样，请看下面的章节）。

这两个选项也可以设置成与 LAN 不相同的配置，如 LAN 192.168.1.1/24，而"服务器地址"和"远程地址范围"分别设置为 192.168.2.254 和 192.168.2.16/28。这样配置，PPTP 连接可以

访问 LAN，但不能防火墙不允许 PPTP 流量路由到 WAN 连接。根据 OPT 和 WiFi 网络的路由设置，它们也将被隔离。

一般情况下，当建立一个 PPTP 连接（尤其是 windows），所有网络流量都将经该 PPTP 隧道收发（除非设置 PPTP 不是缺省网关）

9.5 设置 PPTP

大部分用户可以跳过这一节，如果测试一下面例子遇到问题时，可以回过头来阅读，也许可以找到问题的答案。

1．首先，设置 PPTP 服务器，进入 VPN: PPTP: 配置 页面，参考下图进行设置：



2．选择“启用 PPTP 服务器”

3．输入“服务器地址”，可以在 LAN 中找一个空闲的 IP，要求该 IP 与“远程地址范围”属于相同的子网类型（A，B，C）。

4．输入“远程地址范围”，由于设备最多只支持 16 个并发连接，所以子网规定了只有 16 个 IP 地址（/28）。这里的子网要求与服务器 IP 属于相同的子网类型（但不是在同一个/28 子网中，否则防火墙将提示出错）

在我们的例子中，服务器 IP 使用 192.168.1.254，远程地址范围使用 192.168.1.192/28。当为 PPTP 客户端的 IP（由 VPN 服务器分配）路由时，想象缺省网关是服务器 IP，路由所选择的接口就是与 PPTP 服务器连接的虚拟接口。

如果到这里有不清楚的地方，可以回去阅读：9.4 子网和 VLAN 路由。

5. 如果你有 RADIUS 服务器，可以选择使用，并填写其它输入框。我们的例子不使用（似乎也超出我们的讨论范围）。

6．如果在乎安全，而且客户端也支持，可以选用 128-bit 加密。

7．点击“保存”, 完成 PPTP 服务器的设置。接下来需要设置 PPTP 用户。

9.6 设置 PPTP 用户

如果你有一个 RADIUS 服务器，并且在前一步配置使用它，可以跳过这一节。也可以同时在这里加入用户，以便 PPTP 服务器在向 RADIUS 服务发出请求前，先使用本地的用户进行认证，

点击⊕，可以添加一个新用户：



然后输入用户、密码，也可以为这个用户指定一个 IP：



点击“保存”, 结果：



9.7　PPTP 防火墙规则

通过设置防火墙规则，可以让 PPTP 网络连接更好地工作。就象 LAN 一样，你可以不作任何限制，或根据需要进行限制。通过防火墙规则，可以限制 PPTP 用户能够访问的主机、端口，也可以全开放，不作限制。这里我们假设你允许 PPTP 用户访问整个 LAN，WAN，下面我们将设置一条防火墙规则，就象缺省的 LAN 规则（通行任何数据包）。

1．进入防火墙: 规则页面，点击，增加新规则：



这里我们先添加一条规则，它允许 PPTP 用户访问所有 LAN，WAN，OPT 网络。如果你需要限制对网络的访问，可以根据需要修改规则。下面我们还会给出一个这样的例子。

2．在协议字段，从下拉列表中选择“**Any**”; 在接口字段，从下拉列表中选择**“PPTP”**, 在规则的描述字段最好写上“缺省 PPTP->任何网络”。

3. 点击“保存”, 然后“应用修改”。

这样就完成了 PPTP 服务器的设置。



9.7.1 PPTP 过滤规则的例子

大部分情况下，人们并不完全相信通过 PPTP 接入的用户，需要限制 PPTP 用户的网络访问权限（设置允许通过规则），使用“拒绝”规则阻止其它访问。通过防火规则，可以显式指定允许 PPTP 用户访问的主机、端口，而拒绝其它 IP 流量。例如允许 PPTP 用户访问 LAN，WAN，但不允许他们访问你的 SAMBA 服务器。

我们的例子允许 PPTP 用户通过 SSH 连接 LAN 中 IP 为 192.168.1.151 的服务器。



保存并应用修改，然后测试规则是否按要求工作。很多网络都是由于没有人检查访问许可列表（ACLs），而存在安全隐患。



9.8 设置 windows XP 的 PPTP 客户端

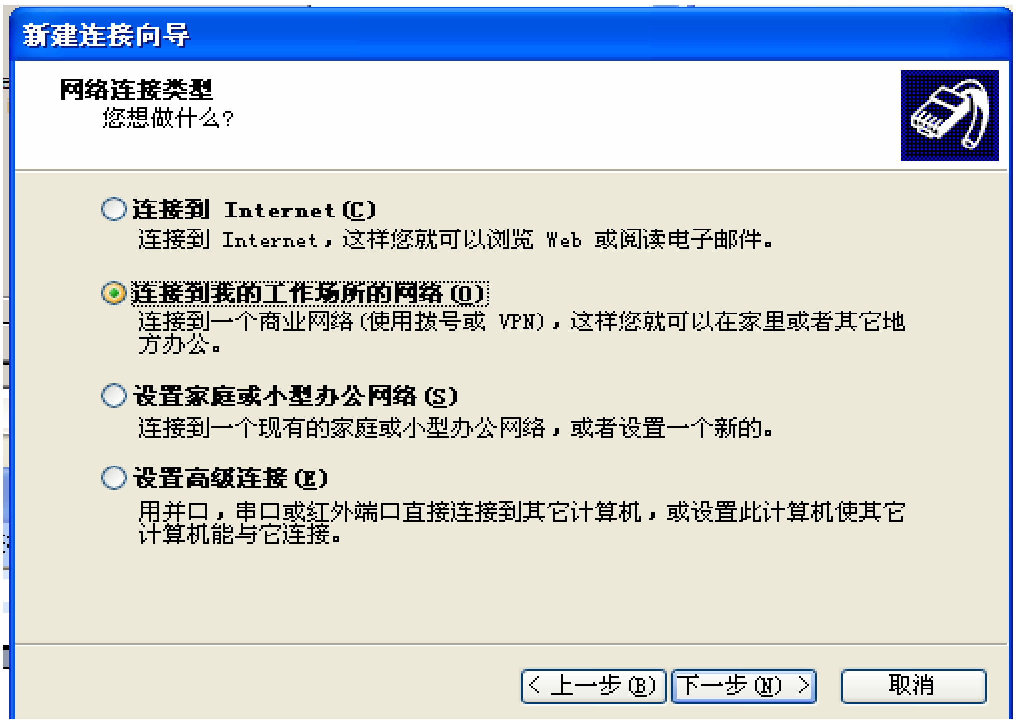
1. 创建新连接



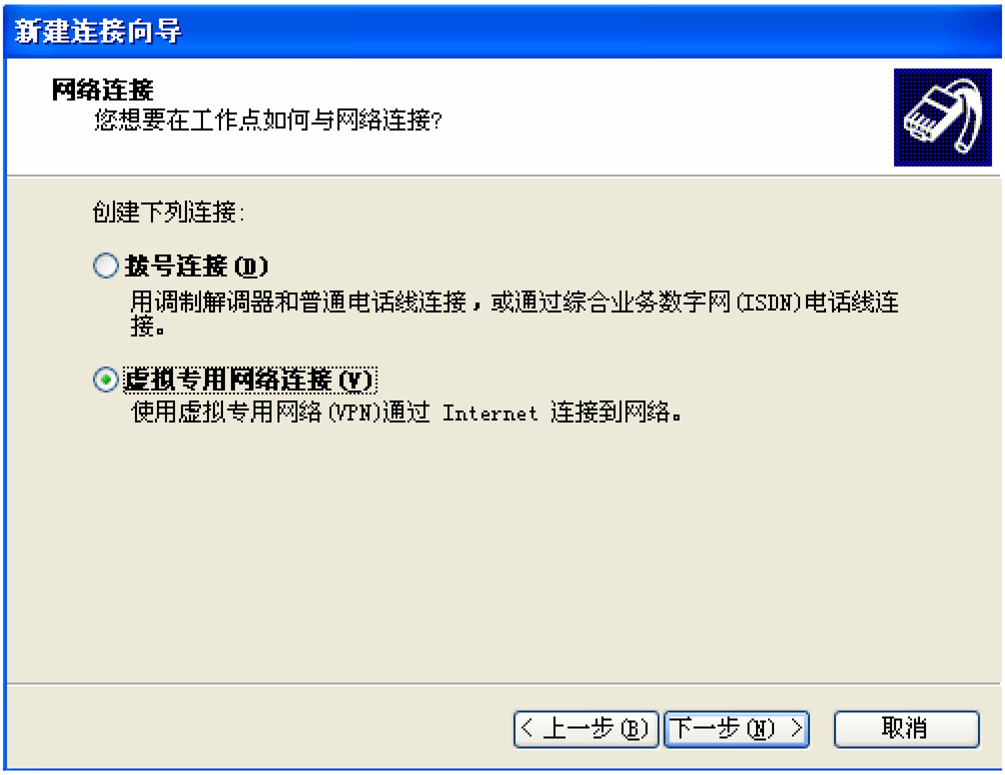
2．点击下一步



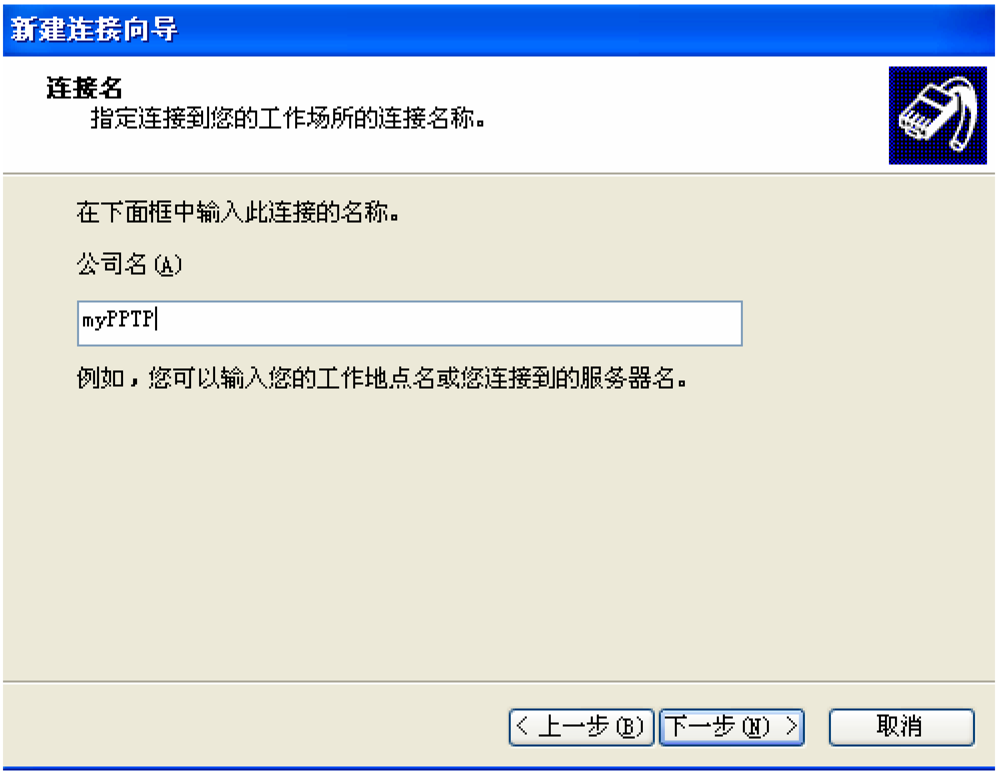
3.选择连接类型：VPN



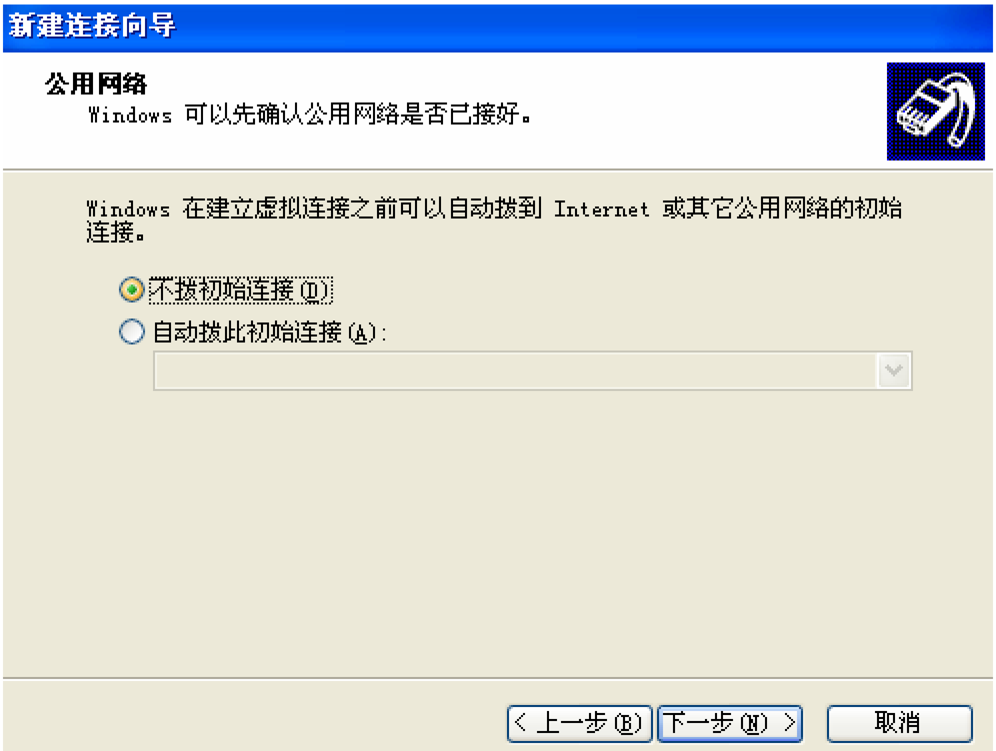
4．选择网络连接方式：VPN



5．输入一个连接名称：如 myPPTP



6. 公用网络：选“不拨初始连接”



7. 输入 PPTP 服务器 IP，这里可以输入服务的域名（可以是动态域名）



点击“下一步”，完成。

8. 测试连接

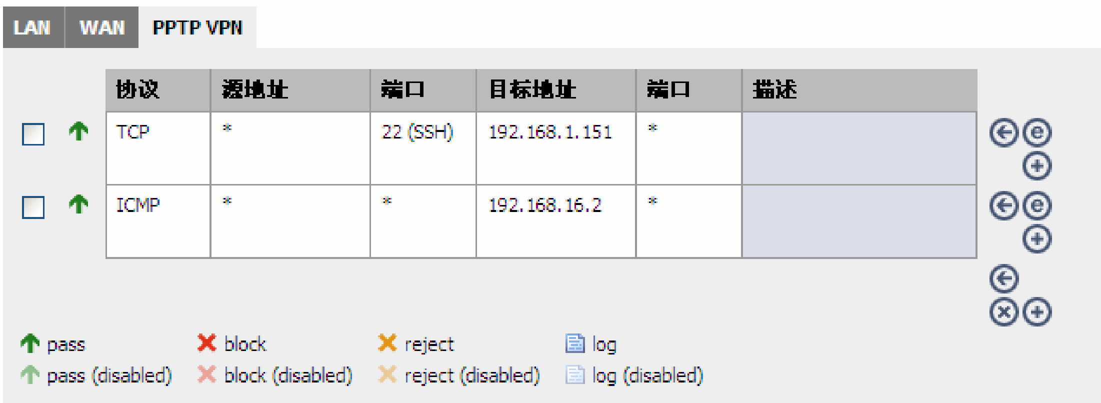


9. 在命令窗口运行 IPCONFIG 检查连接

```
PPP adapter myPPTP:

        Connection-specific DNS Suffix  . :
        IP Address. . . . . . . . . . . . : 192.168.16.192
        Subnet Mask . . . . . . . . . . . : 255.255.255.255
        Default Gateway . . . . . . . . . : 192.168.16.192
```

加入一条允许 PING 通 192.168.16.2 的规则，如图：



协议	源地址	端口	目标地址	端口	描述
TCP	*	22 (SSH)	192.168.1.151	*	
ICMP	*	*	192.168.16.2	*	

测试：

```
C:\>ping 192.168.16.2

Pinging 192.168.16.2 with 32 bytes of data:

Reply from 192.168.16.2: bytes=32 time=1ms TTL=64
Reply from 192.168.16.2: bytes=32 time<1ms TTL=64
Reply from 192.168.16.2: bytes=32 time<1ms TTL=64
Reply from 192.168.16.2: bytes=32 time<1ms TTL=64

Ping statistics for 192.168.16.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 1ms, Average = 0ms
```

9.9 通过 PPTP 连接不能工作的情况

- NAT 与 PPTP 一起使用，有时效果不是很好。m0n0wall 已经对此做了一些处理，它工作得很好。
- 如果你所访问的远程网络与 PPTP 客户端本地网络有相同的网段(网络范围）,这时 PPTP 隧道不能正常工作。例如，你在网吧通过 WiFi 接入互联网，它所分配的 IP 位于 192.168.1.0/24 ,然后你使用 PPTP 连接家里的 PPTP 服务器 ,它也分配一个 192.168.1.0/24 中的 IP，这种情况下，PPTP 隧道可以建立，但 IP 流量不会经 PPTP 隧道收发（由于 TCP 协议栈认为在同一网段，不会把数据包经网关转发）。处理这个问题的方法，可以修改 PPTP 服务分配 IP 的网段（挑选一个很少人使用的）, 或者临时修改 PPTP 客户端机器的 IP 路由表。
- 由于 ISP 使用不合理的 DHCP 租用时间，如一个小时。如果 PPTP 客户端获得一个较短的 DHCP 租用时间，当租约到期时，它将与丢失与互联网的连接。因为所有网络流量，包括 DHCP 更新（renew）请求都需要通过 PPTP VPN 隧道，而这时 DHCP 服务器已经不可到达。解决的办法是手工断开 PPTP 连接，重新拨号。（这种情况可以向有关 ISP 反映，让他们调整 DHCP 的租约时长）。
- M0n0wall 不支持 UPnP

- 由于 PPTP 隧道不转发广播包，Windows 网上邻居将不能正常作。

我对 PPTP 的了解也不是非常多，如果还有更多不能工作的情况，请让我知道，我会把它们加入这个文档，让人们不必把时间花在一些根本就不能工作的问题上。

第十章 OPenVPN

[略]

第十一章 无线接入

[略]

第十二章 Captive Portal

[略]

第十三章 参考

[略]

第十四章 配置例子

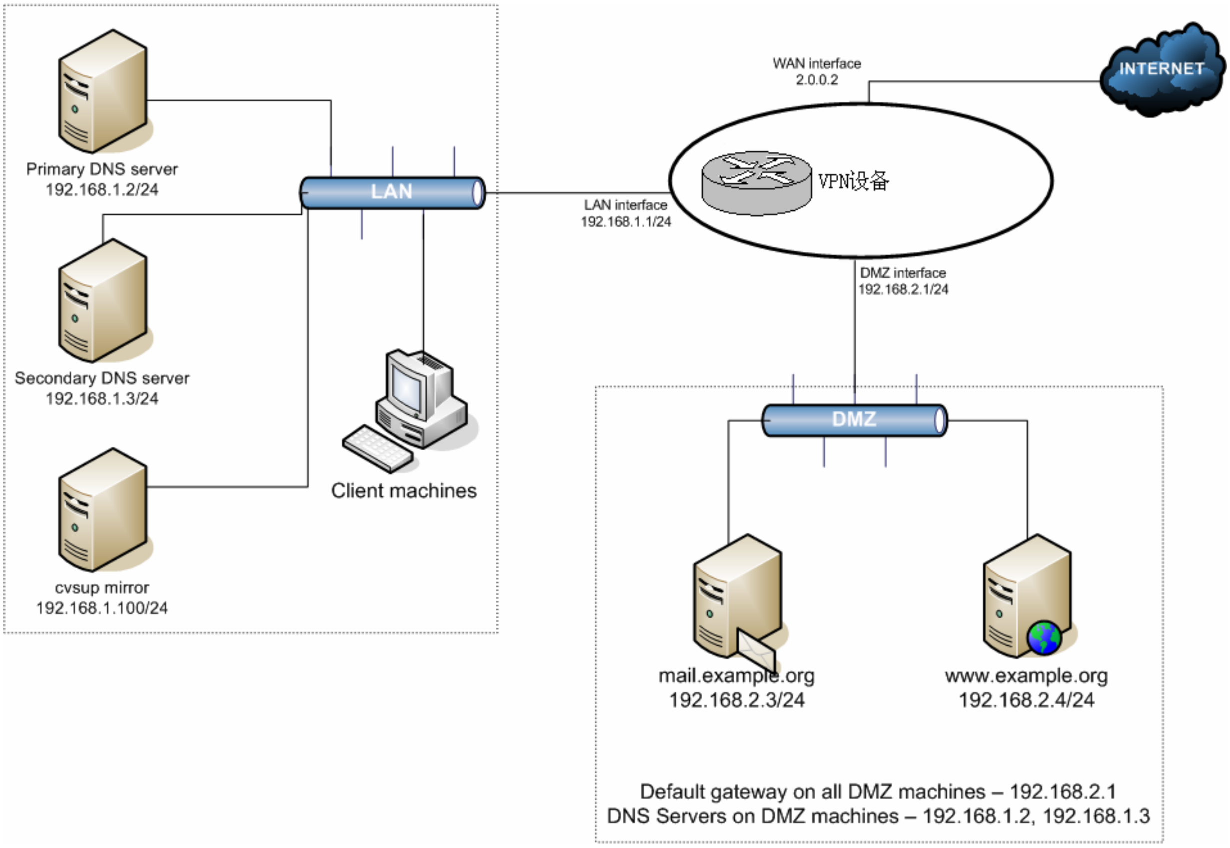
14.1 配置 DMZ 接口使用 NAT

这一节将说明在两个基本网络接口（LAN/WAN）的基础上，如何添加一个DMZ接口。两个网络接口的配置方法参考：Quick Start Guide(http://m0n0.ch/wall/quickstart/)。

在开始配置 DMZ 接口前，应该确保两个基本接口正常工作。

当你有多个公网 IP，而且希望每一台 DMZ 主机对应一个公网 IP，那么使用 1:1 NAT 就很合适。

14.1.1 网络拓扑图



这个图说明了我们完成配置后的网络结构。

14.1.2 添加可选接口

进入接口：分配网络接口页面，点击“”，加入第三个网络接口，其缺省名称为 OPT1，如图：



14.1.3 配置可选接口

在接口菜单中选择 OPT1，如图：



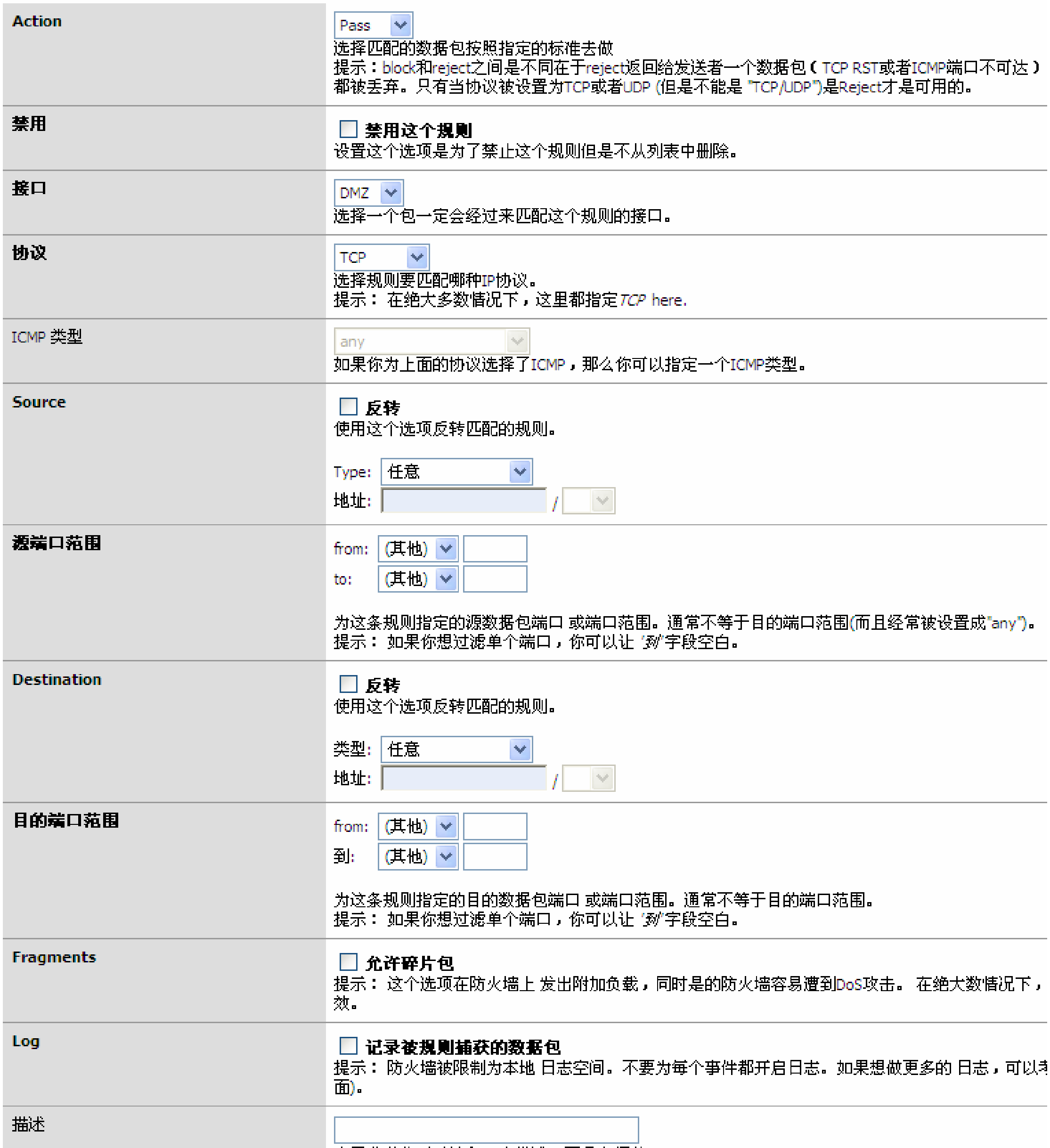
这里选择启用可选接口，把接口名称改为 DMZ，IP 地址设置为 192.168.2.1/24。

14.1.4 配置 DMZ 接口的防火墙规则

设置 DMZ 的主要目的是为了让网络中的 internet 服务主机与 LAN 主机相互隔离，采用这种方法，是为了达到即使 DMZ 中的某一台主机有潜在的安全问题（受到攻击），也不会影响到 LAN 中的其它主机。所以如果不接拦截 DMZ 到 LAN 的 IP 流量，DMZ 基本是没有用的。

进入防火墙：规则页面，选择“DMZ”，点击“⊕”，在 DMZ 接口上添加一条规则：





按照下图填写这个表单，将允许经 DMZ 接口的流量进入 WAN（internet），但禁止所有经

DMZ 的流量进入 LAN。而且规则只允许属于 DMZ 子网的 IP 流量经 DMZ 接口出站，因为只有属于 DMZ 的主机才会经 DMZ 接口出站。这样可以防止地址欺骗。



点击“保存”，然后“应用修改”。

14.1.5 允许服务经 DMZ 进入 LAN

DMZ 中的主机可能需要访问 LAN 中某主机所提供的服务。在我们的样板网络中，DMZ 的主机需要访问 LAN 中的两个 DNS 服务器，使用 cvsup 协议连接 cvsup-mirror 服务器，以及使用 NTP 协议与 TimeServer 同步时间，TimeServer 运行在 cvsup-mirror 服务器。

当定义 DMZ 到 LAN 的规则时，应该明确指出协议、端口、主机，以确保没有不须要的流量通过。

下图是添加规则后样子，它允许所需要的服务通过 DMZ 到达 LAN：

LAN | WAN | PPTP VPN | DMZ

协议	源地址	端口	目标地址	端口	描述
UDP	DMZ net	*	192.168.1.2	53 (DNS)	允许DMZ访问主DNS服务器
UDP	DMZ net	*	192.168.1.3	53 (DNS)	允许DMZ访问辅DNS服务器
TCP	DMZ net	*	192.168.1.100	5999	允许DMZ访问CVSUP服务器
UDP	DMZ net	*	192.168.1.100	123	允许DMZ访问NTP
*	*	*	LAN net	*	拦截经DMZ到LAN的流量
*	DMZ net	*	! LAN net	*	允许DMZ访问除LAN以外的网络

注意到这里加入了一条拦截经 DMZ 进入 LAN 的规则，根据我们所定义的“允许”规则，这是不需要的。这里显式加入这条规则，目的是明确经 DMZ 到 LAN 的流量都会被丢弃。

在输入规则的时候，请记住是按从上到下的顺序处理规则的，而且一旦遇到匹配的规则，就停止处理。所以如果把上述规则放在第一位，则所有流量都会被丢弃，其它“允许”规则不会

被处理。建议参考这里的顺序，把拦截 DMZ 到 LAN 的规则入在倒数第二位，最后允许 DMZ 访问除 LAN 外的其它网络。

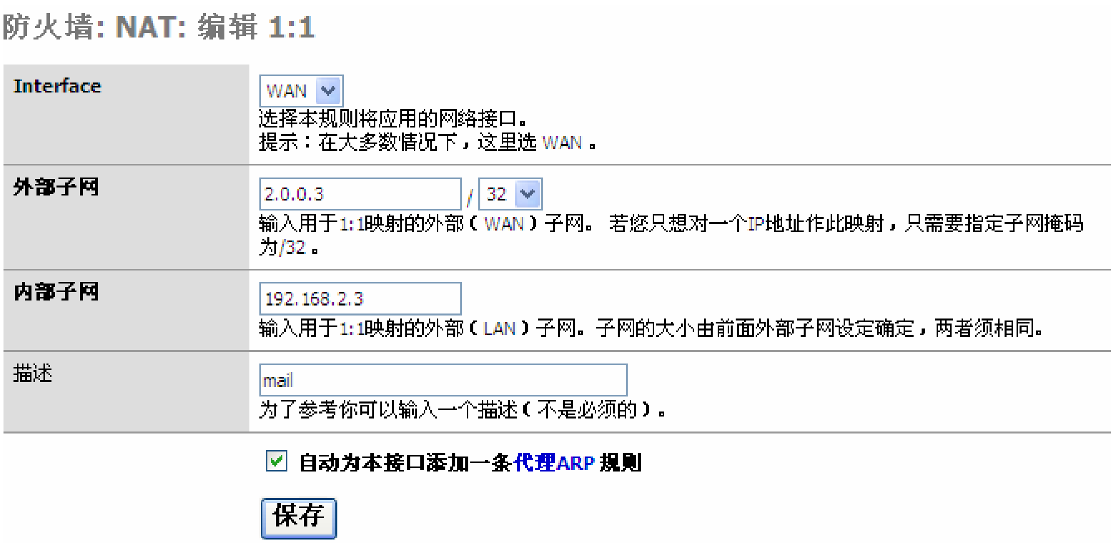
14.1.6 配置 NAT

首先，你需要决定使用转入 NAT（端口映射）还是 1:1NAT。如果你有多个公网 IP，可以使用 1:1NAT。如果你只有一个公网 IP，那么只能使用转入 NAT（端口映射）。如果你有多个公网 IP，但数量比 DMZ 中的主机少，那么可以使用转入 NAT（端口映射），或者结合 1:1NAT 使用。

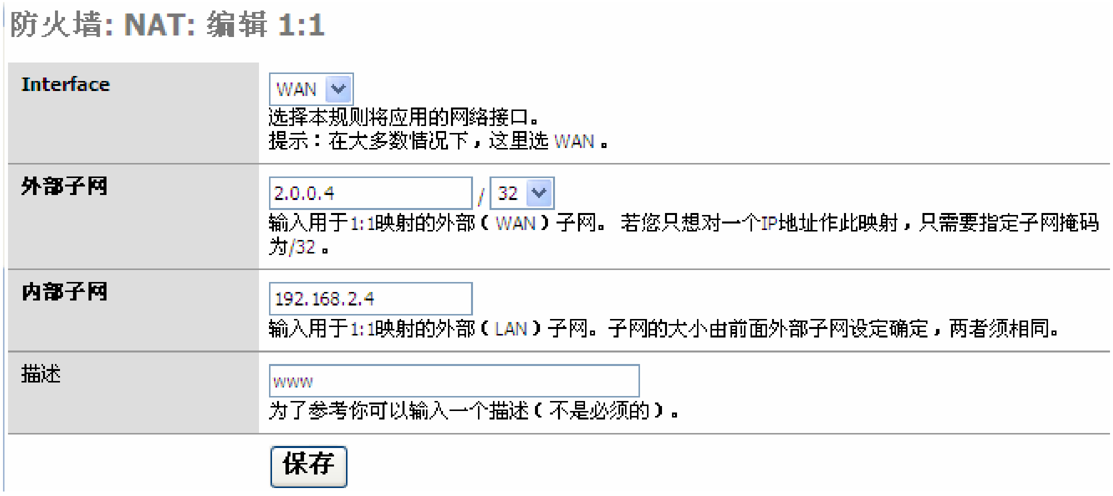
14.1.6.1 使用 1:1NAT

在我们的例子中，我们拥有一个/27 的公网子网，即 2.0.0.0/27。VPN 设备的 WAN 接口已经分配了 IP 2.0.0.2。这里将使用 1:1NAT 把 IP 2.0.0.3 指向 DMZ 中的邮件服务器，IP 2.0.0.4 指向 DMZ 中的 WEB 服务器。

进入防火墙: NAT: 1:1 页面，点击“⊕”。这里将添加两条记录，一条分配给邮件服务器，另一个分配给 WWW 服务器。


防火墙: NAT: 编辑 1:1

Interface	WAN 选择本规则将应用的网络接口。 提示：在大多数情况下，这里选 WAN 。
外部子网	2.0.0.3 / 32 输入用于1:1映射的外部（WAN）子网。 若您只想对一个IP地址作此映射，只需要指定子网掩码为/32 。
内部子网	192.168.2.3 输入用于1:1映射的外部（LAN）子网。子网的大小由前面外部子网设定确定，两者须相同。
描述	mail 为了参考你可以输入一个描述（不是必须的）。
	☑ 自动为本接口添加一条代理ARP 规则 保存



输入的结果如图：



14.1.6.2 测试 1:1NAT 配置

[略]

14.1.6.3 使用转入 NAT（端口映射）

如果只有一个公网 IP，或 DMZ 中需要供公共访问的服务器数量比公网 IP 数量要多，就需要使用转入 NAT（端口映射）。进入防火墙: NAT: 入站页面，选择"转入（端口映射）"，点击"⊕"。

这个例子假设只有一个公网 IP，并且已经分配给了 WAN 接口。

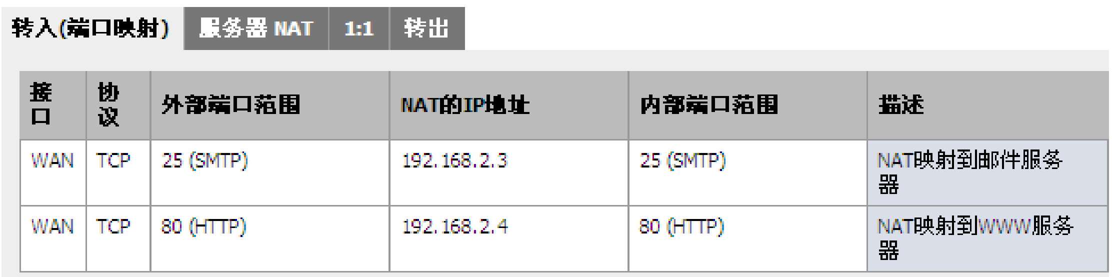
首先，映射 WAN IP，端口 25（SMTP）到 DMZ 中邮件服务器。



点击“保存”, 然后再点击“⊕”, 添加 HTTP 服务器的映射。



点击“保存”，结果如下：



转入(端口映射) | 服务器 NAT | 1:1 | 转出

接口	协议	外部端口范围	NAT的IP地址	内部端口范围	描述
WAN	TCP	25 (SMTP)	192.168.2.3	25 (SMTP)	NAT映射到邮件服务器
WAN	TCP	80 (HTTP)	192.168.2.4	80 (HTTP)	NAT映射到WWW服务器

14.2 限制 DMZ 出站访问

[略]

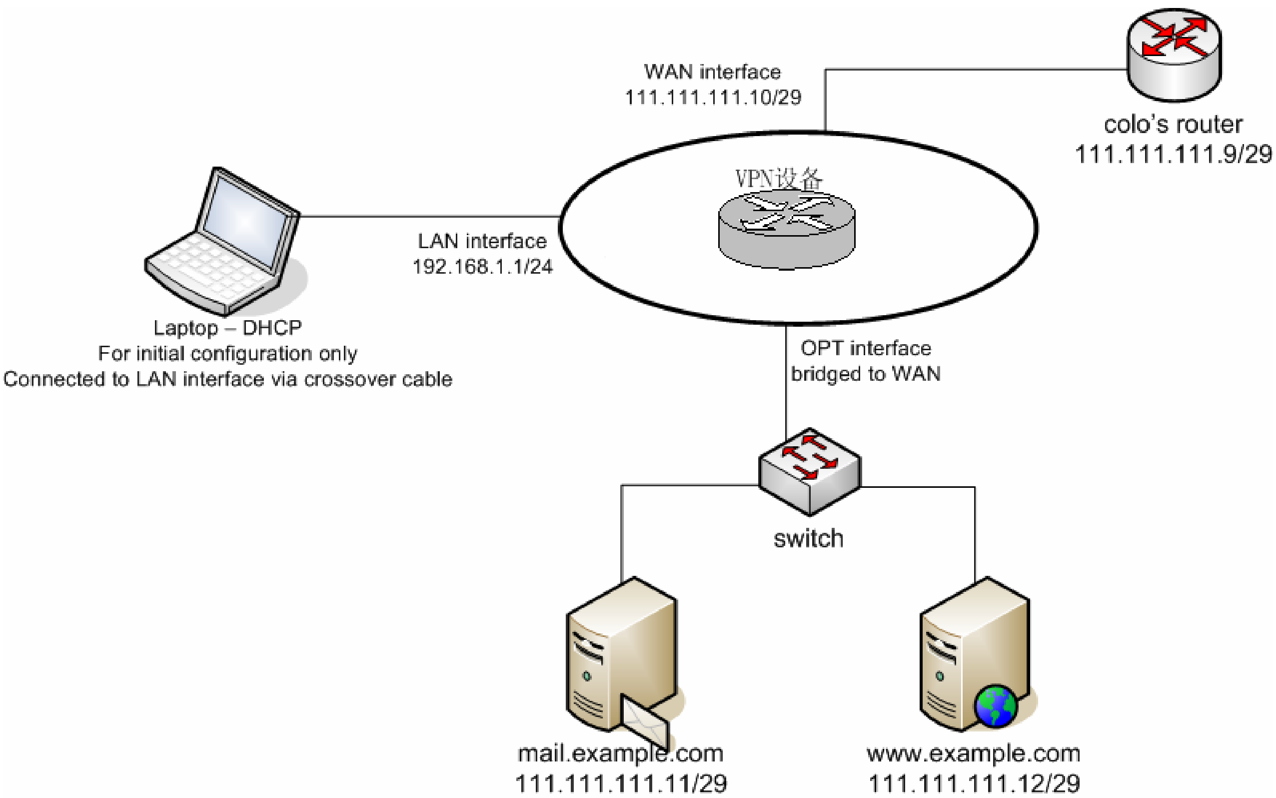
14.3 配置过滤桥接

过滤桥接是常用的 DMZ 配置方法。它可以用在托管机房（colocation facility）保护其中的服务器，VPN 虽然配置了 LAN 接口，但 LAN 中一般没有主机。

提示：

由于从一个"NAT 接口不能直接访问桥接口中的主机",所以，即使配置了有一个 LAN 接口，也不能从 LAN 访问桥接接口后面的主机。

下图是本例的网络拓扑图。图中描述了本例的配置情况，托管机房分配了一个 111.111.111.8/29 的子网（共有 8 个 IP，8 = NOT(0xFFFFFFF8 – 1)），可用 IP 从 9-14（去除了第一个网络 ID 及最后一个广播地址）。其中一个 IP 分配给了路由器，实际可用 IP 只有 5 个。图中的笔记本电脑使用交叉线与 LAN 接口连接。



14.3.1 常规配置

当你按上图准备好网络环境，并且给分配了合适的 LAN IP，登录 webGUI，开始初始配置。

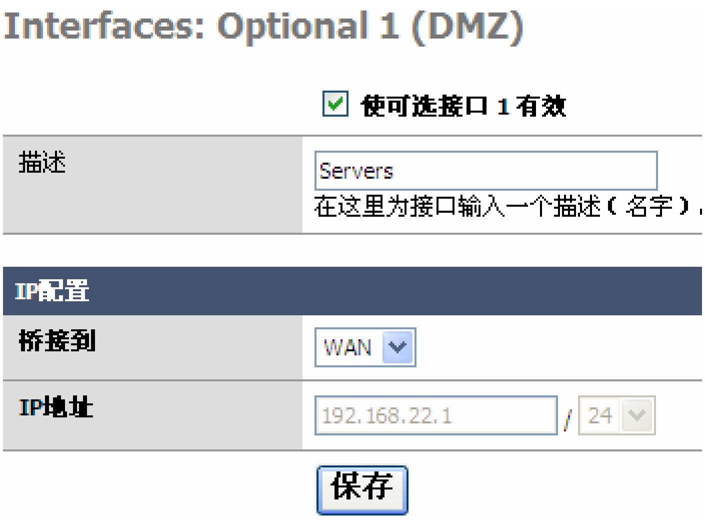
首先进入系统: 常规设置页面，配置主机名，域、DNS 服务器，修改密码，改用 webGUI 改用 HTTPS 方式，选择时区，点击保存，重新启动，让配置生效。

14.3.2 WAN 接口配置

重新登录 webGUI，进入接口:WAN 页面，根据本例的网络配置，这里分配静态 IP111.111.111.10/29，缺省网关为 111.111.111.9。除非你的 WAN 是私有网络，否则应该启用“阻止私有网络”。

14.3.3 OPT 接口配置

选择 OPT 接口，输入你喜欢的名字，这里输入 Servers，在桥接到下拉列表中选择“WAN”(这时 IP 地址字段不能输入)，点击“保存”:



14.3.4 启用过滤桥接功能

进入系统：高级页面，选择“使用桥模式”。点击“保存”。

14.3.5 配置防火墙规则

进入防火墙：规则页面。

> **提示：**
>
> 任何配置都存在漏洞，尤其是限制出站连接。你需要用到更多的规则，只用这里所提及的规则是不够的。开放已知并且需要的连接，同时在防火墙日志中观察被丢弃的数据包，从而发现可能还需要开放的连接。需要花一些时间才能把防火墙限制得尽可能严格，但从长远来说，网络安全相比所花的时间还是值得的。

1.4.3.5.1 OPT 接口规则

初始时，你可以配置一条规则，允许所有流量通过，确认网络可以正常工作后，再根据需要逐步收紧规则。接下来，我们根据本例的要求配置限制规则。

本例中的邮件服务器和 WEB 服务器都位于桥接接口后面，邮件服务器需要向互联网中的任何主机发送邮件。两个服务器都需要访问 DNS 服务器 111.111.110.2 和 111.111.109.2(图中没有标出)。

这里还为 HTTP 和 CVSUP 添加两条禁用的规则，以便于维护。

14.3.5.2 WAN 接口规则

由本例描述防火墙应用于托管机房，我们需要一条用于远程管理的规则，以便允许源自可信静态 IP 的流量可以访问服务器的管理功能，包括 m0n0wall 的 webGUI。本例中，我们允许所有源自可信位置（11.12.13.30）的流量通行。你可以考虑收紧规则，但如果你的 LAN 没有任何主机，那么记住允许访问 webGUI 远程管理 m0n0wall 设备(不必亲自到现场)。

我们还需要添加规则允许 SMTP 流量到达邮件服务器，以及 HTTP/HTTPS 流量到达 WEB 服务器。

14.3.5.3 LAN 接口规则

如果 LAN 中没有其它主机，那么可以删除缺省规则或留着它不管。在本例中，当完成现场配置后，与 LAN 连接的机器会被撤走。

14.3.5.4 完成配置的规则

LAN | WAN | PPTP VPN | Servers

		协议	源地址	端口	目标地址	端口	描述
	✕	*	RFC 1918 networks	*	*	*	Block private networks
☐	↑	TCP	11.12.13.30	*	*	*	允许远程管理
☐	↑	TCP	*	*	111.111.111.11	25 (SMTP)	允许SMTP到达邮件服务器
☐	↑	TCP	*	*	111.111.111.12	80 (HTTP)	允许HTTP到达WEB服务器
☐	↑	TCP	*	*	111.111.111.12	443 (HTTPS)	允许HTTPS到达WEB服务器

LAN | WAN | PPTP VPN | Servers

协议	源地址	端口	目标地址	端口	描述
TCP	111.111.111.8/29	*	*	80 (HTTP)	HTTP维护规则
TCP	111.111.111.8/29	*	*	5999	CVSUP 维护规则
TCP	111.111.111.8/29	*	111.111.110.2	53 (DNS)	允许访问NS1
TCP	111.111.111.8/29	*	111.111.109.2	53 (DNS)	允许访问NS2
TCP	111.111.111.11	*	*	25 (SMTP)	允许邮件服务到达任何服务器

14.3.6 完成配置工作

到这里，如果服务器都配置正确，一切都应该按预期地工作。测试出站、入站规则是否按照预期的要求工作。当所有测试结果都达到要求，配置就完成了。

第十五章 site to site VPN 配置的例子

M0n0wall 可以与任何支持标准 IPsec site-to-site VPN 的第三方设备互联，包括大部分支持 IPsec 的 VPN 和防火墙设备。

这一章给出 m0n0wall 与部分第三方设备互相连接的设置指导。

如果你成功配置m0n0wall与一个设备互联，而该设备没有在这里列出，请写出你的设置方法。

（There is a section of the wiki dedicated to configurations for this chapter.）

15.1 Cisco PIX Firewall

这一节说明 m0n0wall 与 PIX 防火墙之间建立 IPsec 隧道的方法。

15.1.1 PIX 的配置

首先确认 PIX 已经启用了 3DES。

```
pixfirewall# sh ver

Cisco PIX Firewall Version 6.3(3)
Cisco PIX Device Manager Version 2.0(2)

Compiled on Wed 13-Aug-03 13:55 by morlee

pixfirewall up 157 days 5 hours

Hardware: PIX-515E, 32 MB RAM, CPU Pentium II 433 MHz
```

如果“VPN-3DES-AES”的值不是显示为“Enable”,那么需要安装PIX 3DES key。现在所有的PIX防火墙都可以从CISCO取得3DES Key（http://www.cisco.com/pcgi-bin/Software/FormManager/formgenerator.pl，点击3DES/AES Encryption License）。为了密码安全，VPN连接不要使用DES。DES仅仅比传输明文要好一些。

接下来，需要查看 PIX 中是否已经有其它 VPN 配置。

```
pixfirewall# sh isakmp policy


Default protection suite

encryption algorithm: DES - Data Encryption Standard (56 bit

keys).
```

如果只看到缺省的策略，说明没有 VPN 的配置。如果已经有 VPN 配置存在，那么操作时就不能完全按照这里的说明来做（但你应该可以自己处理，关键是注意不要使用与现有 VPN 配置重复的名字）

允许与 PIX 建立 IPSec 连接：

```
pixfirewall(config)# sysopt connection permit-ipsec
```

在 outside 接口启用 ISAKMP(outside 接口指与 internet 连接的接口):

```
pixfirewall(config)# isakmp enable outside
```

先查看 PIX 中 isakmp policy 命令的格式：

```
pixfirewall(config)# isakmp policy ?

Usage: isakmp policy %lt;priority> authen %lt;pre-share|rsa-sig>

isakmp policy %lt;priority> encrypt %lt;aes|aes-192|aes-256|des|3des>

isakmp policy %lt;priority> hash %lt;md5|sha>

isakmp policy %lt;priority> group %lt;1|2|5>

isakmp policy %lt;priority> lifetime %lt;seconds>
```

现在配置 PIX 中的 ISAKMP 策略，在配置模式输入以下命令:

```
isakmp policy 10 authen pre-share

isakmp policy 10 encrypt 3des

isakmp policy 10 hash md5

isakmp policy 10 group 2

isakmp policy 10 lifetime 86400
```

这个策略使用预共享密钥（pre-shared keys）, 3DES 加密，MD5 HASH，生存期 86400 秒。

接着需要定义这个 VPN 连接的预共享密钥（pre-shared keys）, 其中 1.1.1.1 是 m0n0wall

```
isakmp key qwertyuiop address 1.1.1.1 netmask 255.255.255.255
```

的公网 IP，qwertyuiop 是共享密钥，在你的实际配置中，可以随机生成。

```
access-list monovpn permit ip 10.0.0.0 255.255.255.0 10.0.1.0 255.255.255.0
access-list monovpn permit ip 10.0.0.0 255.255.255.0 10.0.1.0 255.255.255.0
```

现在定义访问列表，定义能够通过 VPN 隧道的流量。

```
crypto ipsec transform-set monovpnset esp-3des esp-md5-hmac
```

定义传输方式，命名为："monovpnset"

定义安全联盟（SA）的生存期：

```
crypto ipsec security-association lifetime seconds 86400 kilobytes 50000
```

现在设置真正的 VPN 连接(crypto map)，命名为"monovpnmap", 其中 1.1.1.1 是 m0n0wall 的公网 IP。

```
crypto map monovpnmap 10 ipsec-isakmp
crypto map monovpnmap 10 set peer 1.1.1.1
crypto map monovpnmap 10 set transform-set monovpnset
```

这里指定 VPN 的类型（ipsec-isakmp）, 对端 IP（1.1.1.1）, 使用的传输加密方式(monovpnset, 在上面定义的)，以及与访问列表 monovpn(在上面定义)匹配的数据包可以经这个 VPN 连接传输。

最后还要告诉 PIX，对于经 VPN 传输的数据包，不要使用 NAT，而应该使用路由方式来处理。先查看一下 NAT 的当前路由设置：

```
pixfirewall# sh nat
nat (inside) 0 access-list no-nat
```

查看“nat (inside) 0…”命令 ,上面显示的结果意思是 ,任何流量只要与“no-nat”访问列表匹配，

```
access-list no-nat permit ip 10.0.0.1 255.255.255.0 10.0.1.0 255.255.255.0
access-list no-nat permit ip 10.0.1.0 255.255.255.0 10.0.0.0 255.255.255.0
```

将被路由，而不进行 NAT。本例将向现有访问列表增加规则（如果你使用 DMZ，那么你的配置与这个例子就很相似）。

如果你的“sh nat”没有“nat (inside) 0…”输出 ,你仍然可以使用上面两条命令建立一个“no-nat”访问列表。然后使用以下命令启用“no-nat”访问列表：

```
nat (interface-name) 0 access-list no-nat
```

实际操作时，需要用具体的 LAN 接口名称替换“interface-name”。

15.1.2 m0n0wall 的配置

登录 webGUI，在 VPN 菜单选择 IPSec。

如果“启动 IPSec”没打钩，钩上，然后保存。

点击“+”按钮，添加新的 IPSec 隧道定义，各项参数设置如下：

使这个隧道无效：不打钩

接口：WAN

NAT-T：不打钩（如果设备确实在 NAT 后面请钩上）

本地子网：选“LAN 子网”

远端子网：10.0.0.0/24（这是指位于 PIX 后面的子网，根据实际情况填入）

远端网关：PIX 的公网 IP

描述：如“PIX VPN”

阶段 1

协商模式：Aggressive

我的标识 :“My IP Address”

加密算法：3DES

HASH 算法：MD5

DH Key Group: 2

生存期：86400

预共享密钥：qwertyuiop (输入与 PIX 中一样的密钥)

阶段 2

协议：ESP

加密算法：**只选择** 3DES

HASH 算法：**只选择** MD5

PFS key group: 2

生存期：86400

提示：

如果使用 m0n0wall 1.2 beta 版本，使用这个配置时你可能会遇到连接经常中断，如果是这样，把 PFS key group 设置为:OFF

提示：

如果在 m0n0wall 的配置中不指定生存期，隧道可能在工作一段时间后变得不通畅。推测可能是 Cisco 会协商一个生存期限，但在实际看到这个过程。Cisco 的 VPN concentrator 也有这种情况。

15.2 Smoothwall

15.3 FreeS/WAN (OpenSwan)

15.4 Sonicwall

15.5 Nortel

第十六章 常见问题问答(FAQ)